U.PORTO

Universidade do Porto
FEUP Faculdade de Engenharia

Projeto de automação de máquina de gravação LASER para pneus do tipo PCR

Miguel Ferreira da Cunha Barros

Dissertação de Mestrado
Faculdade de Engenharia da Universidade do Porto
Mestrado Integrado em Engenharia Mecânica
Ramo de Automação

Versão final

Orientadores:
Gislótica: Eng. Michel Bizarro
Gislótica: Eng. Humberto Ramos
FEUP: Prof. António Pessoa de Magalhães

Porto, Julho 2021

Resumo

Os pneus atuais contêm marcações na sua parede lateral com variadas informações técnicas. A empresa Gislotica - *Mechanical Solutions* levou a cabo o projeto mecânico de uma máquina que realizasse este tipo de marcações através de tecnologia LASER em pneus do tipo PCR.

A máquina consistirá numa cabine individual e duas estações para transferir pneus de e para esta cabine. Dentro da mesma, estão colocados todos os equipamentos mecânicos necessários para realizar o processo de gravação. A parte do desenvolvimento denominada projeto de automação, que inclui seleção de componentes e programação, constituiu o objetivo principal desta dissertação.

Visto que se trata da primeira versão de uma máquina, esta servirá como prova de conceito. No entanto, para efeitos de dissertação, foi imposto que deveria ser desenvolvida como se fosse para integrar numa fábrica com alimentação automática por um sistema de transportadores, trocasse parâmetros de processo com um servidor SQL e utilizasse PLC e SCADA da marca Rockwell Automation.

Numa primeira fase, é feita uma apresentação dos vários componentes que constituem a máquina, do processo a realizar, uma previsão dos vários movimentos durante o ciclo de funcionamento e dos modos de funcionamento que se pretendem implementar. Segue-se o dimensionamento e seleção de componentes que será divido em dois passos: primeiro do sistema de potência e depois do sistema de comando.

É então que se passa à programação do controlador. A lógica, que é desenvolvida segundo certas linhas gerais traçadas pela empresa, é apresentada de forma independente do ambiente de desenvolvimento utilizado. É explicado o método de controlo dos vários componentes e a lógica de estados.

A secção final desta dissertação consiste na descrição da visualização criada para o SCADA, através da análise dos ecrãs produzidos.

Inicialmente, previa-se que a máquina fosse construída à medida que o segundo semestre do ano letivo de 2020/2021 se desenrolava para que se pudessem realizar arranques iniciais, o que não aconteceu. Não obstante, todos os objetivos foram cumpridos e o código devidamente testado utilizando meios alternativos para simular o processo.

Palavras-chave: Gravação LASER, Máquina Industrial, Gravação de pneus, Programação PLC, Rockwell Automation, Servomotor, FactoryTalk, Studio5000.

Abstract

Some technical information about a given tyre can be found by reading the markings on its sidewall. Gislotica - Mechanical Solutions has designed a machine that is capable of LASER printing such markings in PCR tyres.

The machine is made up of a central enclosed station and two conveyor units for transporting tyres into and from the former. Inside the central station is where the process of printing occurs. The scope of this dissertation is to design this machine in terms of industrial automation through the selection of components, programming the machine's controller, and creating the user interface.

Since this is the first version of this machine, this is merely a proof of concept. However, it should be developed as if it were to be put in a factory, with an automatic conveyor line, an SQL server for receiving process parameters and using a PLC and SCADA by Rockwell Automation.

The dissertation starts with the presentation of the machine, the process of printing, the planned movement of each component, and the operating modes to be implemented. Secondly, commercial models for certain automation components regarding power and logic will be chosen.

After that, we proceed to the description of the PLC logic implemented, independently of the development environment that was used. This consists of properly approaching each component's interface individually and describing the controller's state logic.

Finally, the SCADA visualization is described through the presentation of each screen that was created.

The machine was predicted to be built during the second semester of 2020/2021 so that the code could be tested. However, since the machine was not built in time, the code had to be tested using simulators so that every objective was fulfilled.

Keywords: LASER Engraving, Industrial Machine, Tyre engraving, PLc Programming, Rockwell Automation, Servomotor, FactoryTalk, Studio5000.

Agradecimentos

No âmbito desta dissertação, gostaria de expor os meus agradecimentos começando pelos meus orientadores na Gislotica: os Engenheiros Humberto Ramos e Michel Bizarro. Os seus conhecimentos e capacidade de os transmitir foram extremamente úteis, e certamente me levaram a encontrar uma solução melhor. A meu ver, ambos são profissionais que servem de exemplo a qualquer estudante que os tenha como orientadores.

Ainda da Gislotica, quero agradecer aos Engenheiros Rui Fazenda e Paulo Silva que me deram todas as condições necessárias para trabalhar e estiveram sempre prontos a ajudar-me com dúvidas que surgissem. Não esquecendo obviamente todos os colaboradores da empresa com quem contactei e me integraram num contexto industrial, sempre com muita simpatia.

Quero expressar também o meu agradecimento ao Professor António Pessoa de Magalhães, que desde o dia em que aceitou a minha proposta de tese, se revelou sempre disponível para me acompanhar. A sua atenção e ajuda ao longo de todo o percurso contribuíram muito positivamente para o resultado final.

Ao Professor Francisco Freitas, pela sua prontidão em introduzir-me à Gislotica e por todos os conhecimentos que me passou dentro e fora das salas de aula.

Um obrigado aos meus amigos que me acompanharam durante este semestre em momentos de trabalho e convívio de forma equilibrada, que foram bastante importantes durante esta etapa final.

À minha namorada Verónica Monteiro, por me acompanhar e compreender a minha dedicação à engenharia.

Por último, um agradecimento especial aos meus familiares que sempre me acompanharam incondicionalmente e me deram a oportunidade de seguir este caminho profissional.

Muito obrigado a todos.

Índice de Conteúdos

Resumo **iii**

Abstract **v**

Agradecimentos **vii**

1 Introdução **1**
1.1 A Gislotica - Projeto e Fabrico de Soluções Mecânicas, Lda 2
1.2 Motivação para a realização deste projeto 3
1.3 Objetivos da dissertação 3
1.4 Organização do documento 4

2 Apresentação da máquina **7**
2.1 Fluxo do processo de Marcação de um pneu 7
2.2 Composição da máquina 9
2.3 Funcionamento pretendido 14
2.4 Considerações sobre a segurança da máquina 17
2.5 Síntese do capítulo 20

3 Seleção de componentes **21**
3.1 Componentes do circuito de potência 21
3.1.1 Seleção de servomotores e caixas redutoras 21
3.1.2 Seleção de componentes elétricos do circuito de potência . . . 29
3.2 Componentes do circuito de comando 38
3.2.1 PLC 38
3.2.2 Codificador para a câmara 40
3.2.3 *Splitter* do sinal do codificador 42
3.2.4 Módulo de codificador alternativo 43
3.2.5 Computador Industrial 44
3.2.6 *Software* de SCADA 46
3.2.7 Sensores fotoeléctricos 47
3.2.8 Barreira de medição 49
3.3 Síntese do capítulo 50

4 Desenvolvimento do programa de controlo da máquina **53**
4.1 Introdução aos controladores atuais da Rockwell Automation 53
4.2 Considerações sobre os componentes da máquina 57

4.2.1 Elementos de comando da HMI 61
4.2.2 Interface com o LASER 62
4.2.3 Comunicação através de *sockets* TCP/IP 65
4.2.4 Interface com a câmara 67
4.2.5 Interface com o transportador e Multicontrol 70
4.2.6 Interface com o PLC de linha 72
4.2.7 Interface com o servidor SQL 74
4.2.8 Interface com cilindros pneumáticos 80
4.2.9 Interface com eixos servomotorizados 80
4.3 Descrição comportamental do funcionamento da máquina 82
4.3.1 *Grafcet* 1 - Controlo do modo automático 84
4.3.2 *Grafcet* 0 - Controlo central 86
4.3.3 *Grafcet* 2 - Rotina de reinicialização 87
4.3.4 *Grafcet* 3 e 5 - Ciclos automáticos das estações de entrada e saída 88
4.3.5 *Grafcet* 4 - Ciclo automático da estação central 90
4.4 Aplicação prática e testes 95
4.4.1 Organização de programas e rotinas do controlador 96
4.4.2 Programação de suporte a situações de erro 98
4.4.3 Soluções utilizadas para simular o processo 99
4.5 Criação dos ecrãs do SCADA 101
4.5.1 Painel de cabeçalho 105
4.5.2 Ecrã *Main* 106
4.5.3 Ecrã *Machine Components* 108
4.5.4 Ecrã *Settings* 110
4.5.5 Ecrã *Inputs & Outputs* 112
4.5.6 Ecrã *Alarms & Events* 112
4.6 Síntese do capítulo 113

5 Conclusão e trabalhos futuros 115
5.1 Conclusão 115
5.2 Trabalhos Futuros 116

Referências bibliográficas 119

A Cálculos realizados para a seleção de servomotores e caixas redutoras 123
Referências bibliográficas do anexo A 147

B Desenho esquemático da posição do pneu durante o processo 149

C Ecrãs criados no FactoryTalk SE para interface com o operador 151

Lista de Figuras

2.1 Diagrama de fluxo do processo. 8
2.2 Disposição espacial das três estações da máquina. 10
2.3 Renderização de uma "*Tire Flipping Station*", uma obra Gislotica. Renderização: Gislotica. 11
2.4 Componentes no interior da estação central. Renderização: Gislotica. 12
2.5 Renderização de uma possível configuração final da máquina. Renderização: Gislotica. 14
2.6 Diagrama de Gant do processo de gravação. 16
2.7 Disposição dos componentes de segurança da máquina. 18

3.1 Sistema de acionamento correia-polia. Projeto: Gislotica. 23
3.2 Perfil temporal de posição (traço contínuo verde) e velocidade (traço interrompido azul) a considerar. 24
3.3 *Drives* de motores considerados de ambas as marcas. 30
3.4 Detalhe das entradas de segurança (STO) de um *drive* de RA. 31
3.5 Perfil temporal do binário - T_{carga} (traço contínuo) e velocidade - n_{carga} (traço interrompido) e potência - P_{carga} (traço - ponto) da carga do eixo B. Fonte: SIZER da Siemens, editado. 34
3.6 Controlador da gama CompactLogix 5380, com alguns módulos de I/O encaixados. 39
3.7 Esquema da ligação feita entre o codificador, a câmara e o PLC. . . . 41
3.8 Exemplo de uma transmissão do tipo correia - polia entre um veio e um codificador. 42
3.9 *Splitter* GV-204 da Motrona [23]. 43
3.10 Módulo de saída de encoder [24]. 44
3.11 Computador industrial AIIS-3400P (PoE) [27]. 45
3.12 Monitor da gama Versaview 5100 [29]. 47
3.13 Esquema de um sensor fotoelétrico retrorrefletivo. 48
3.14 Sensor retrorefletivo da Sick pertencente à família GL-10 [35]. 48
3.15 Barreira de medição da Banner pertencente à família MLG-2 [36]. . . 50

4.1 Organização hierárquica de tarefas, programas e rotinas num controlador. 55
4.2 Dispositivos que interagem com o PLC da máquina durante o processo. 60
4.3 LASER da Koenig & Bauer que será utilizado na máquina. Modelo: SPA - C 10. 62

4.4 Esquema da colocação da câmara relativamente ao pneu. 67
4.5 Legenda dos componentes externos de uma Gocator [20]. 68
4.6 Campo de visão de uma Gocator. 69
4.7 Módulo Multicontrol da Interrol para controlo de rolos. 71
4.8 Diagrama de estados da transferência de pneu entre um transportador da máquina e um transportador da fábrica. 74
4.9 Esquema das etapas de transferência de um pneu. 76
4.10 Diagrama hierárquico de interação entre *grafcets*. Linhas azuis: *Feedback*. Linhas cor-de-laraja: ordens. 84
4.11 *Grafcet* 1 - Controlo do modo automático. 85
4.12 *Grafcet* 0 - Controlo central. 86
4.13 *Grafcet* 2 - Rotina de reinicialização. 88
4.14 89
4.15 *Grafcet* 4 - Ciclo automático da estação central. 91
4.16 Expansão da macroetapa M1 - Rotina de verificação da largura e diâmetro externo do pneu. 92
4.17 Expansão da macroetapa M2 - Rotina para determinação da posição da referência do pneu. 93
4.18 Expansão da macroetapa M3 - Rotina de gravação a LASER na parede do pneu. 94
4.19 Expansão da macroetapa M4 - Rotina de abortagem de um pneu. . . 95
4.20 *Printscreen* do Studio 5000 onde se vê a organização de programas na *MainTask*. 97
4.21 *Printscreen* do Studio 5000 onde se vê a organização de subrotinas em cada programa. 98
4.22 Cenário criado no Factory IO para simulação do processo. 100
4.23 *Printscreen* do terminal do simulador do LASER. 101
4.24 Esquema de navegação entre os ecrãs principais no FactoryTalk. . . . 104
4.25 Painel de cabeçalho. 106
4.26 Ecrã *Main*. 107
4.27 Painel com informação sobre a estação central. 108
4.28 Painel com informação sobre a rotina de reinicialização. 108
4.29 Ecrã *Machine Components*. 109
4.30 Ecrã *Machine Components*. 109
4.31 Ecrã *Seetings*, aba *Recipe*. 110
4.32 Ecrã *Seetings*, aba *Connections*. 111
4.33 Ecrã *Alarms & Events*. 112

A.1 Componentes que constituem o eixo A. Projeto: Gislotica. 124
A.2 Perfil temporal de posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo A. Fonte: SIZER da Siemens. . 125
A.3 Perfil temporal do binário - T_{polia} (traço contínuo) e velocidade angular - ω_{polia} (traço interrompido) da polia do eixo A. Fonte: SIZER da Siemens. 126
A.4 Perfil temporal da posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo B. Fonte: SIZER da Siemens. . 130

A.5 Perfil temporal do binário - T_{polia} (traço contínuo) e velocidade angular - ω_{polia} (traço interrompido) na polia do eixo B. Fonte: SIZER da Siemens. 132
A.6 Perfil temporal da posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo C. Fonte: SIZER da Siemens. . 136
A.7 Perfil temporal do binário na polia - T_{polia} (traço contínuo) e velocidade angular da polia - ω_{polia} (traço interrompido) do eixo acionado pelo servomotor C. Fonte: SIZER da Siemens. 137
A.8 Componentes que constituem o eixo D. Projeto: Gislotica. 141
A.9 Perfil temporal da aceleração - a_{cel} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo D. Fonte: SIZER da Siemens. 142
A.10 Perfil temporal do binário - T_{motor} (traço contínuo) e velocidade angular - ω_{servo} (traço interrompido) do servomotor D. Fonte: SIZER da Siemens. 143
A.11 Curva de funcionamento do servomotor D. Fonte: SIZER da Siemens. 145

Lista de Tabelas

3.1 Servomotores e caixas redutoras escolhidos para as soluções Siemens e RA. 28
3.2 Correntes efetiva e máxima em cada servomotor e o respetivo módulo de motor escolhido da marca Siemens. 35
3.3 Características dos Drives escolhidos da marca Siemens. 36
3.4 Correntes efetiva e máxima em cada servomotor e o respetivo módulo de motor escolhido da marca RA. 37
3.5 Características dos Drives escolhida para a marca RA. 37

4.1 Elementos físicos de controlo no painel da HMI. 61
4.2 Mensagem enviada do PLC para o LASER a pedir as horas do seu relógio interno. 63
4.3 Mensagem enviada do LASER para o PLC a responder ao pedido das horas do seu relógio interno, neste exemplo é dia 6/11/1998 e são 15:55:59. 64
4.4 Parâmetros de entrada e saída da câmara. 70
4.5 Entradas e Saídas digitais trocadas com o PLC de linha e o respetivo significado quando apresentam o valor *True*. 73
4.6 Membros da variável *status* do pneu. 82
4.7 Membros da variável receita do pneu. 83

A.1 Características técnicas da caixa redutora escolhida para o eixo A. . . 128
A.2 Características técnicas do servomotor Siemens escolhido para o eixo A. 129
A.3 Características técnicas do servomotor Rockwell Automation escolhido para o eixo A. 129
A.4 Características técnicas da caixa redutora escolhida para o eixo B. . . 134
A.5 Características técnicas do servomotor Siemens escolhido para o eixo B. 135
A.6 Características técnicas do servomotor Rockwell Automation escolhido para o eixo B. 135
A.7 Características técnicas da caixa redutora escolhida para o eixo A. . . 139
A.8 Características técnicas do servomotor Siemens escolhido para o eixo A. 140
A.9 Características técnicas do servomotor Rockwell Automation escolhido para o eixo A. 140

A.10 Características técnicas da caixa redutora escolhida para o eixo D. . . 142
A.11 Características técnicas do servomotor Siemens escolhido para o eixo D. 144
A.12 Características técnicas do servomotor Rockwell Automation escolhido para o eixo A. 146

Acrónimos

ADO *ActiveX Data Object*.

ADODB *ActiveX Data Object - Data Base*.

AGVs *Automated Guided Vehicles*.

AoI *Add-on Instruction*.

API *Application Programming Interface*.

CIP *Common Industrial Protocol*.

COM *Component Object Model*.

CPU *Central Processing Unit*.

DC *Duty Cycle*.

DLL *Dynamic Link Library*.

ETRMA European Tyre and Rubber Manufacturers' Association.

HMI *Human-Machine Interface*.

HTL *High Transistor Logic*.

IDE *Integrated Development Environment*.

IP *Internet Protocol*.

IPC *Industrial Personal Computer*.

LASER *Light Amplification by Stimulated Emission of Radiation*.

LED *Light Emitting Diode*.

ME *Machine Edition*.

MES *Manufacturing Execution System*.

ODBC *Open Database Connectivity*.

OLEDB *Object Linking and Embedding - Data Base.*

OPC-UA *Open Platform Communications - Unified Architecture.*

OTR *Off The Road.*

PAC *Programmable Automation Controller.*

PCR *Passenger Car Radial.*

PL *Performance Level.*

PLC *Programmable Logic Controller.*

QR *Quick Response.*

RA Rockwell Automation.

RAM *Random Access Memory.*

SCADA *Supervisory Control and Data Acquisition.*

SE *Site Edition.*

SFC *Sequential Function Chart.*

SLAT *Speed-Limited Adjustable Torque.*

SQL *Structured Query Language.*

SS1 *Safe Stop 1.*

SS2 *Safe Stop 2.*

SSD *Solid State Drive.*

ST *Structured Text.*

STO *Safe Torque Off.*

TCP *Transmission Control Protocal.*

TTL *Transistor-Transistor Logic.*

UDP *User Datagram Protocol.*

UDTs *User-defined Data Types.*

VBA *Visual Basic for Applications.*

Capítulo 1

Introdução

A roda é reconhecida por muitos como a maior invenção do homem. A roda mais antiga conhecida pertencia a uma carroça e estima-se que date da idade do cobre, contando com mais de cinco mil anos. Por um lado, o conceito mudou pouco: um objeto cilíndrico que gira em torno do seu eixo e rola sob uma superfície, garantindo geometricamente que o seu eixo translade linearmente.

Por outro lado, a roda primitiva - toda ela feita do mesmo material maciço - podia ser melhorada: ao utilizar outro material protetor da sua superfície, conseguiu-se estender o seu tempo de vida, e gozar das propriedades únicas da cada um dos materiais onde cada um era necessário.

No ano de 1845, o inventor Escocês Robert William Thomson desenvolveu um daqueles que hoje são conhecidos como pneumáticos (ou pneus) [1; 2]. Na altura, o pneu era constituído por uma câmara de ar em borracha da índia protegida por couro. Os anos correram, a ciência e a engenharia avançaram, e a roda nunca foi deixada para trás: a humanidade tratou de massificar esta grande invenção.

Os pneus comercializados hoje em dia são grandes obras da engenharia e quase todos seguem filosofias semelhantes: uma estrutura altamente resistente ao desgaste no exterior e um interior oco onde é aprisionado e pressurizado um fluído (normalmente ar atmosférico), com ou sem recurso a câmaras de ar. A estrutura é heterogénea em termos de materiais, sendo utilizados vários ingredientes tais como: borracha natural, sintética, negro de fumo, sílica e aço [3], que levantam questões de caráter ambiental e ético, que devem ser devidamente abordadas, mas ficam para lá do alcance deste documento.

O crescimento exponencial da produção de pneus deve-se muito ao crescimento da indústria automóvel. O mercado global dos pneus foi estimado em 2019 com o valor de $ 112,16 biliões, e com uma previsão de crescimento de 4.1% por ano até 2027 [4], sem ter em conta o impacto da pandemia COVID-19. Como tal, os grandes fabricantes de pneus têm que estar na vanguarda da tecnologia para conseguir dar uma resposta competitiva e tecnologicamente capaz de ir ao encontro das necessidades dos consumidores.

O panorama pandémico devido à doença infeciosa COVID-19 resultou na suspensão temporária de atividades nas fábricas de grandes e médios produtores em todo o globo. Os produtores pertencentes à ETRMA (European Tyre and Rubber Manufacturers' Association) registaram no final do ano de 2020 quebras nas ven-

das na Europa de pneus de origem para veículos de passageiros na ordem dos 23% [5]. Esta queda é uma consequência direta de uma queda na indústria automóvel também devido à pandemia, e espera-se que haja uma recuperação com ajuda de instituições europeias e governos de outras nacionalidades.

A maior parte das etapas do processo de produção em série de um pneu requer a utilização de máquinas industriais. Controversamente, são cada vez mais procurados pneus específicos para uma dada aplicação. Assim, é necessário flexibilizar e modularizar a produção em série para que seja possível na mesma fábrica produzir pneus diferentes, sem perder tempo a alterar o parque de máquinas.

O aspeto visual de um pneu é o seu primeiro impacto com um possível cliente, pelo que é necessário que a sua qualidade seja visualmente percecionável. A marcação de um pneu é executada no lado exterior da sua parede lateral e consiste em informação técnica, tal como processo de fabrico, características nominais, método de montagem, utilização prevista e produtor. Tradicionalmente, os pneus são marcados no processo de cura [6]; contudo, se dois pneus curados na mesma prensa precisarem de diferentes marcações, é preciso recorrer a outras máquinas.

É exatamente neste contexto que se insere a máquina-alvo desta dissertação: máquina de gravação a *Light Amplification by Stimulated Emission of Radiation* (LASER) na parede de pneus do tipo *Passenger Car Radial* (PCR). À data do início desta dissertação, o projeto mecânico desta máquina já havia sido desenvolvido pela Gislotica e estava em processo de produção e construção nas suas oficinas. O objetivo desta dissertação é realizar o seu projeto de automação, que consiste na seleção de componentes e programação.

1.1 A Gislotica - Projeto e Fabrico de Soluções Mecânicas, Lda

A Gislotica é uma empresa de soluções de engenharia para a indústria dos pneus que se localiza em Perafita, no Porto.

No que toca a máquinas, a Gislotica leva a cabo o seu projeto, fabrico mecânico, automação, instalação e manutenção, fornecendo aos seus clientes soluções chave-na-mão e assegurando o seu funcionamento e qualidade. Seja por encomenda de um cliente ou por motivação interna, as soluções são criadas de acordo com uma visão de valor para a indústria, promovendo a inovação e avanço tecnológico. Em alguns projetos, é feita uma parceria com outras empresas especializadas na área como forma de garantir o melhor serviço aos seus clientes.

A empresa persegue a sua missão: garantir uma qualidade, que a identifique junto dos seus clientes. Para isto, usufrui do conhecimento, experiência e recursos comprovados para garantir soluções integradas para a indústria. Alguns exemplos de projetos são linhas de transporte de pneus de vários tamanhos, estações de centragem de pneus na linha de produção e sistemas de transporte automático de arame.

A política de qualidade da empresa é um assunto de grande importância, já que terá consequências em situações em que é necessário decidir entre várias soluções possíveis. A empresa procura que as soluções transpareçam o tempo de estudo e desenvolvimento dedicado, assegurando resultados excelentes e sustentabilidade.

Isto é conseguido através da escolha de soluções robustas, inovadoras, criativas, que cumpram os requisitos legais e regulamentares aplicáveis, garantindo ao mesmo tempo o cumprimento dos requisitos do cliente, a melhoria contínua da qualidade e uma presença competitiva no mercado.

É nesta empresa que vários estudantes e aprendizes realizam os seus projetos de fim de ciclo e adquirem experiência profissional. São vários os estudantes do Departamento de Engenharia Mecânica da Universidade do Porto que aqui realizam as suas dissertações de mestrado.

1.2 Motivação para a realização deste projeto

Tradicionalmente, as marcações resultam da colocação do pneu em verde num molde e aplicação de pressão e calor. A geometria destas marcações está gravada no próprio molde e é transferida para o pneu em verde durante esta etapa.

Este método é ótimo para tipos de marcações inerentes ao método de fabrico, ou seja, que sejam iguais para todos os pneus produzidos da mesma maneira. Com o emergir da customização em massa, cada vez existe maior necessidade de fazer marcações diferentes em dois pneus com processos de fabrico que apenas divergem em pequenos aspetos. Assim, seria útil que fosse marcado um código de série único em cada pneu, ou então, por exemplo, um código *scaneável* digital que redirecione para um endereço da *internet*.

Outro aspeto importante das marcações é a sua qualidade, que deve ser suficientemente boa para que não comprometa a perceção da qualidade do produto marcado. Por exemplo, a rugosidade superficial de um logótipo, ou dos rótulos "*Sport*" e "*Efficiency*" podem ser importantes em termos de apelo ao cliente. Adicionalmente, a gravação de códigos digitais, tais como os códigos *Quick Response* (QR) também requer alto contraste para permitir a leitura correta dos bits codificados.

A gravação a LASER é capaz de realizar este tipo de marcações, produzindo figuras complexas de forma rápida e de elevada qualidade. A empresa viu no mercado uma oportunidade de desenvolver uma máquina para fazer este tipo de gravações em pneus, apostando numa solução de projeto mecânico inovadora.

Em máquinas concorrentes, o pneu entra na máquina e é centrado; o LASER roda em torno de um eixo, posicionando-se na posição desejada do pneu para efetuar a gravação. Na máquina que se apresentará, o pneu é que irá rodar em vez de ser o LASER, eliminando a dobragem cíclica dos cabos e tornando o deslocamento angular entre pneu e LASER praticamente ilimitado. Assim, numa primeira fase, esta máquina servirá como prova de conceito de uma tecnologia, e não como uma simples solução para vender imediatamente a um cliente.

1.3 Objetivos da dissertação

O desafio proposto consiste na seleção de componentes de potência e de comando para uma máquina de gravação a LASER e posterior desenvolvimento do programa para o controlador da máquina e do respetivo *Supervisory Control and Data Acquisition* (SCADA). Nesta secção serão descritos em detalhe cada um dos objetivos

especificados na proposta da dissertação de mestrado.

A máquina contém cinco eixos servomotorizados, para realizar certos movimentos importantes para o processo. O primeiro objetivo foi selecionar os servomotores a implementar em cada eixo, das marcas Siemens e Rockwell Automation (RA). Para tal, fez-se uma previsão dos ciclos da máquina e calcularam-se várias métricas para determinar a sua exigência.

Para cada servomotor selecionado, procedeu-se ao dimensionamento e seleção dos componentes elétricos de acionamento, nomeadamente *drives* e fontes de alimentação. Para auxiliar a seleção, recorreu-se ao *software* de dimensionamento destes componentes de ambas as marcas. Não obstante, os cálculos necessários foram todos feitos à parte e serão devidamente apresentados, para garantir que não se cometeram erros de utilização.

Depois, com base no interesse de um possível cliente, decidiu-se enveredar apenas pelo desenvolvimento da solução para uma das marcas, a RA. Foi feita a seleção de um controlador para a máquina, *software* de SCADA e de outros componentes que fazem parte da automação de uma máquina. Isto foi conseguido através da consulta de manuais e catálogos de fabricantes sugeridos pela empresa.

Selecionados todos os componentes, procedeu-se ao desenvolvimento de um programa de raiz para o controlador da máquina, de acordo com as especificações dadas pela empresa. Esta operação foi levada a cabo no ambiente de desenvolvimento da RA, o Studio 5000. A consulta de manuais e livros técnicos, fóruns *online* e discussão com outros profissionais da área foi crucial para atingir este objetivo.

A máquina deve conter uma *Human-Machine Interface* (HMI) com um ecrã onde deve corre uma visualização do SCADA da RA, o FactoryTalk, e um conjunto de botoneiras físicas. É através da HMI que o operador comanda e configura a máquina. O objetivo seguinte é desenvolver a aplicação do SCADA, que passa pela criação de vários ecrãs funcionais.

O objetivo final estava previsto como "testes e arranque inicial da máquina", contudo a máquina não pôde ser construída atempadamente. Para dar a volta ao problema, todos os programas foram devidamente testados em simuladores e bancas de teste montadas especialmente para este fim. Todos os meios utilizados para teste foram reconhecidos pela empresa como alternativas que validam este objetivo.

1.4 Organização do documento

Esta dissertação está dividida em cinco capítulos e três anexos, servindo o presente como introdução.

O capítulo 2 começa com a apresentação do projeto previamente realizado pela empresa do qual o autor partiu no início desta dissertação. Nesse sentido, são abordados o fluxo de acontecimentos que devem ter lugar no processo, componentes que constituem a máquina e modos de funcionamento previstos. Seguidamente, apresentam-se as extrapolações feitas pelo autor para prever a sequência de ações que deve decorrer ao longo de um ciclo completo de funcionamento automático e considerações sobre elementos de segurança da máquina.

O capítulo 3 diz respeito ao dimensionamento e seleção de modelos comerciais de componentes de automação da máquina. Primeiramente, será feita a escolha de

servomotores, caixas redutoras e componentes de acionamento do circuito elétrico, através da análise de ciclos de funcionamento. Numa segunda fase, serão selecionados elementos de controlo e sensorização, tais como o controlador da máquina e sensores.

O capítulo 4 tem como objetivo expor a programação realizada do controlador da máquina e da visualização desenvolvida para correr na HMI. Começar-se-á por fazer uma breve introdução do funcionamento dos controladores da RA, abordando depois aspetos como a lógica de controlo de cada um dos componentes da máquina e lógica de ciclo .

Ainda no capítulo 4, como toda a programação desenvolvida foi testada num PLC, adicionou-se uma secção que explica em que medidas é que isto foi realizado. Na secção final, são apresentados e analisados os ecrãs principais do SCADA.

O capítulo 5 encerra a dissertação. É neste que se incluem as conclusões gerais do autor relativamente a todos os assuntos tratados. A dissertação termina com a referência aos trabalhos futuros necessários para que a máquina fique no estado chave-na-mão, pronta para começar a gravar pneus.

O primeiro anexo contém informação detalhada sobre os eixos servomotorizados presentes na máquina, para os quais foi necessário fazer seleção de servomotores e caixas redutoras. Nesse anexo, são ainda apresentados todos os cálculos necessários e modelos comerciais selecionados.

No anexo B é incluído um esquema da posição assumida pelo pneu durante as etapas de gravação a LASER e de reconhecimento da sua superfície através da câmara de visão. Será utilizado para explicar o calculo da posição dos vários eixos da máquina durante o processo.

Por último, o anexo C contém todos os ecrãs criados (principais e secundários) do SCADA que constituem a visualização a correr no monitor da HMI.

Capítulo 2

Apresentação da máquina

Este capítulo tem como objetivo expor os elementos que constituem a máquina e o respetivo papel na marcação da parede (externa) de um pneu. O capítulo engloba os seguintes tópicos:

- Análise do fluxo do processo que se pretende realizar para a gravação a LASER de um pneu;
- Apresentação dos componentes físicos que constituem a máquina, fruto do projeto mecânico já realizado pela empresa;
- Descrição dos modos de funcionamento pretendidos;
- Descrição do ciclo de gravação a LASER, e dos movimentos que devem ser realizados;
- Apresentação das considerações feitas sobre segurança de máquinas;

Finda a leitura, estarão assentadas as bases para proceder aos capítulos seguintes.

2.1 Fluxo do processo de Marcação de um pneu

O objetivo desta secção é expor ao leitor a sequência lógica de etapas que têm de ser percorridas para poder efetuar uma operação de gravação na parede de um pneu. Nos parágrafos seguintes, será apresentado o fluxo do processo de gravação (figura 2.1).

A empresa especificou que se trata de uma máquina para integrar numa fábrica que faça processamento dos pneus em série. Isto significa que os pneus circulam numa linha de produção que os alimenta um a um, através de um transportador automático. Os pneus circulam deitados, isto é, com uma parede voltada para a superfície do transportador e outra para cima e não estão perfeitamente posicionados ao centro do transportador.

A gravação pode-se querer fazer em ambas as paredes, e como tal deve-se prever a possibilidade de fazer uma viragem do pneu em torno de um eixo horizontal, perpendicular à direção de circulação do transportador. A existência desta etapa de

viragem constitui uma decisão do cliente, em função da qual a empresa adaptará a máquina para cada caso em particular.

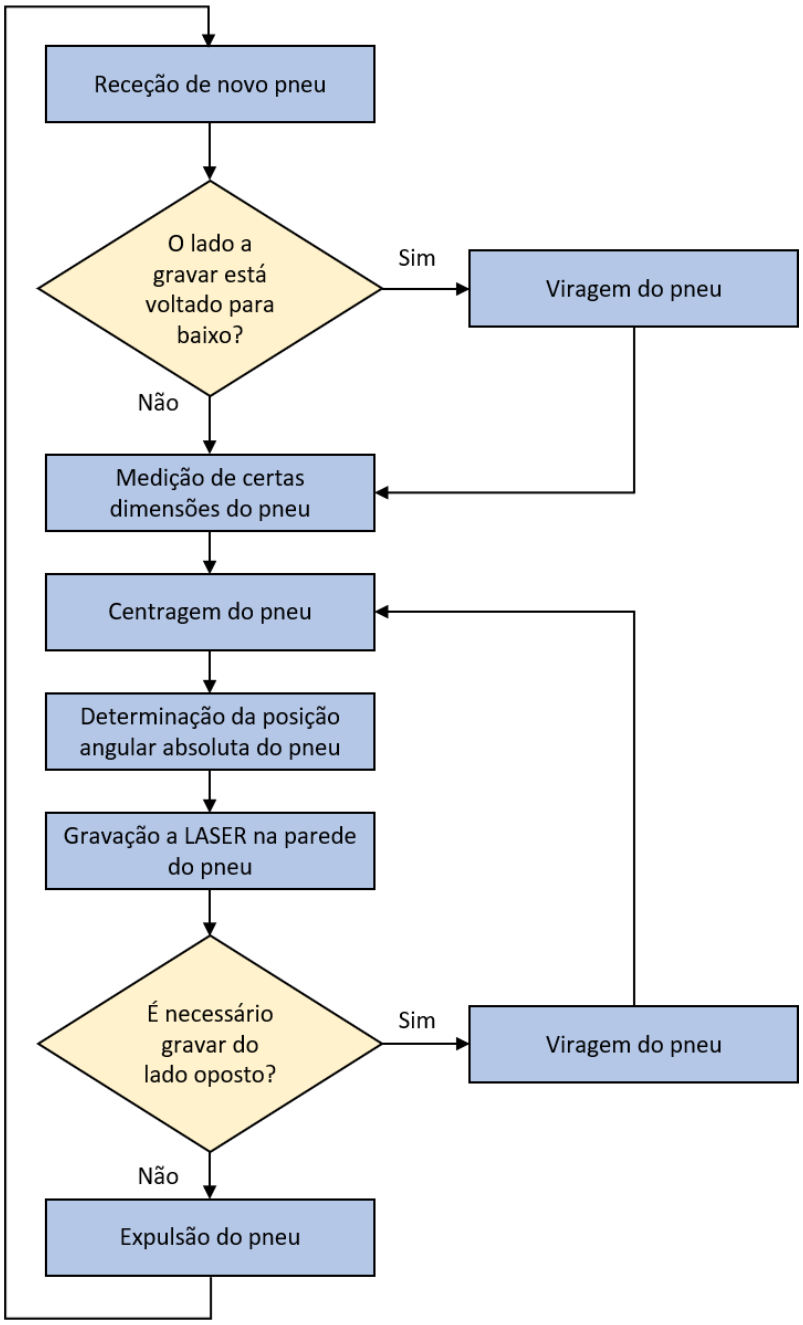


Figura 2.1: Diagrama de fluxo do processo.

A cada pneu está associada uma receita em formato digital, com informações necessárias para o processo e que deve ser interpretada pelo controlador da máquina. Independentemente da maneira como a máquina obtém a receita, existe a possibilidade de esta estar errada ou não corresponder ao pneu que entra na máquina. Como tal, o cliente pode optar pela implementação na máquina de uma maneira para verificar algumas dimensões físicas do pneu, garantindo que receita está correta.

Para as etapas seguintes do processo, é necessário que o pneu esteja centrado no

transportador porque a sua posição deve estar bem definida. Como o transporte por meio de transportadores não garante precisão da posição do pneu quer na direção do fluxo, quer na direção perpendicular ao fluxo, é necessário centrá-lo. Desta forma, a posição do pneu passa a estar bem definida, permitindo proceder às etapas seguintes.

A gravação deve ser feita numa posição específica do pneu, definida por um diâmetro e por um ângulo (definido em coordenadas polares por se tratar de um objeto de revolução). A referência da variável angular consiste numa pequena marca no pneu com uma certa geometria, mas a sua posição é desconhecida quando o pneu entra na máquina. Para a determinar, é preciso utilizar um sistema de visão que percorre toda a superfície da parede do pneu, detetando a referência e determinando a sua posição angular relativamente a um referencial cohhecido.

Conhecida a posição angular exata do pneu, pode-se proceder à gravação do mesmo no ponto desejado. Para tal, é utilizado um sistema de gravação a LASER comprado à parte e instalado na máquina. Feito isto, existem duas hipóteses: a primeira é expelir o pneu da máquina e passar ao seguinte. A segunda opção é virar o pneu, gravar na parede oposta e só depois expeli-lo e passar ao pneu seguinte.

2.2 Composição da máquina

Nesta secção apresenta-se como é constituída a máquina considerada para efeitos desta dissertação, fazendo um apanhado geral dos componentes mecânicos importantes. A descrição detalhada do funcionamento e mecanismos de certos componentes, será deixada para os capítulos subsequentes.

No âmbito desta dissertação, considera-se que a máquina foi encomendada para ser implementada numa fábrica cuja alimentação dos pneus é feita automaticamente de um transportador da fábrica para o transportador da máquina, sem qualquer interação de terceiros[1]. Estes transportadores são controlados por um PLC presente na fábrica, responsável pela linha de produção.

A receita do pneu, que deverá conter as suas dimensões nominais e parâmetros para a gravação, é obtida através da consulta de um servidor *Structured Query Language* (SQL) presente na rede da fábrica. Quando um pneu estiver para entrar na máquina, deverá ser acrescentada uma linha a uma tabela específica do servidor com a respetiva chave primária, a pedido do PLC da fábrica; após a receção do pneu, esta linha deverá ser eliminada, a pedido do PLC da máquina. O processo inverte-se ao expelir um pneu, recorrendo a uma outra tabela.

Adicionalmente, é necessário implementar verificação de duas dimensões dos pneus recebidos: a largura e o diâmetro externo. A empresa especificou que o pneu apenas deve ser gravado do lado em que é recebido, não sendo necessária nenhuma etapa de viragem nesta variante da máquina. No entanto, deve ser tomada especial atenção para que a inclusão de etapas de viragem seja de fácil implementação no futuro, de maneira modular.

A máquina é constituída por três estações, que se definem como:

1. Estação de entrada;

[1] Com terceiros está-se aqui a referir à possível interação humana, de *Automated Guided Vehicles* (AGVs), robôs ou outro sistema que consiga manipular um pneu no espaço.

2. Estação central;

3. Estação de saída

Tal como o nome indica, as três estações referidas são dispostas em série, da maneira representada na figura 2.2. O pneu entra na máquina pela estação de entrada, é processado na estação central e finalmente expelido de volta para a fábrica na estação de saída.

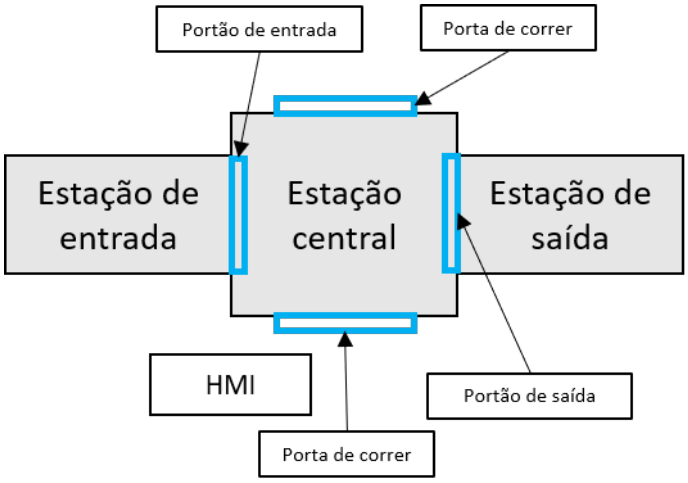


Figura 2.2: Disposição espacial das três estações da máquina.

As estações de entrada e saída servem de interface da máquina com a fábrica, e são constituídas apenas por um transportador de rolos com sensores em ambas as extremidades. Nestes transportadores, apenas pode existir um pneu de cada vez, não funcionando como acumuladores[2].

Na versão da máquina com capacidade de fazer a viragem do pneu, seriam colocadas umas estações que constituem uma obra à parte da empresa denominadas "*Tire Flipping Station*" (figura 2.3). Estes viradores permitem fazer rotações do pneu em torno de um plano horizontal, caso se queira fazer gravação do lado oposto daquele em que o pneu se encontra. Reforça-se, no entanto, que a viragem do pneu não faz parte do processo e não são utilizados estes viradores.

A estação central da máquina é constituída por todos os elementos responsáveis por efetuar a gravação propriamente dita do pneu. Na figura 2.4 estão representados todos os elementos no interior da estação central que se passam a enumerar:

- Servomotores:
  - ⇒ Eixo A - Movimento de centragem do pneu;
  - ⇒ Eixo B - Movimento vertical do conjunto que transporta o LASER e a câmara;
  - ⇒ Eixo C (LASER) - Movimento horizontal do LASER;
  - ⇒ Eixo C (Câmara) - Movimento horizontal da câmara;

[2]Transportadores acumuladores são aqueles em que é possível acumular pneus através do movimento individual de certos conjuntos de rolos, que não é o objetivo destas estações.

⇒ Eixo D - Rotação da mesa para posicionar o pneu.

- Cilindros pneumáticos:
  - ⇒ Portão de entrada;
  - ⇒ Portão de saída;
  - ⇒ Elevação da mesa.
- Câmara de visão para conhecer a posição angular do pneu;
- LASER para fazer a marcação do pneu;
- Transportador de rolos;
- Mesa elevatória rotativa;
- Quatro sensores fotoelétricos.



Figura 2.3: Renderização de uma "*Tire Flipping Station*", uma obra Gislotica. Renderização: Gislotica.

Como se pode ver pela figura 2.4, os servomotores são utilizados para o acionamento mecânico de cinco eixos; uma explicação detalhada dos mecanismos de cada eixo pode ser consultada no anexo A. Os dois primeiros cilindros pneumáticos são utilizados para abrir ou fechar os portões de entrada do pneu; estes abrem de cima para baixo e não podem estar sempre abertos devido à reflexão da radiação LASER emitida.

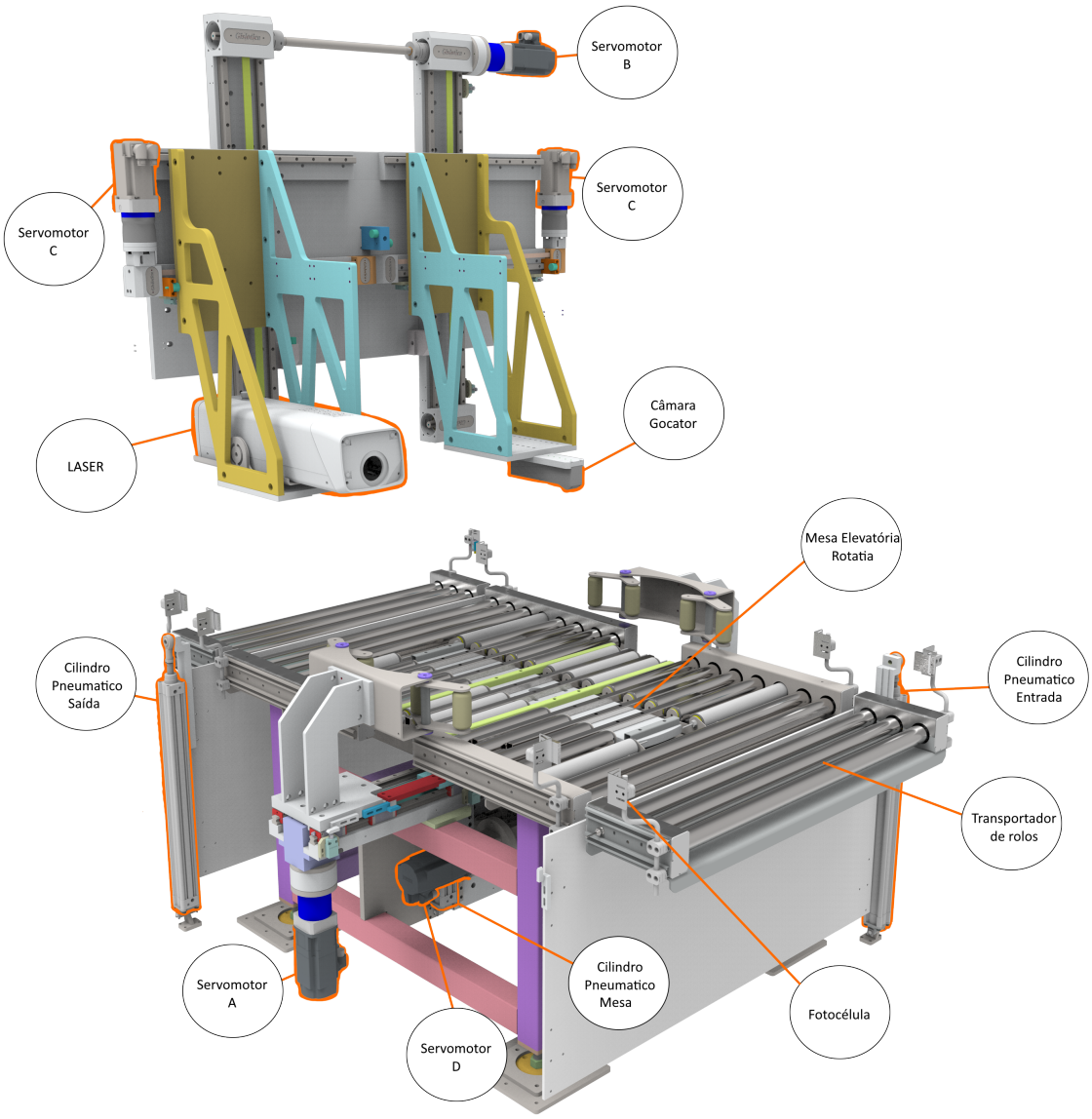


Figura 2.4: Componentes no interior da estação central. Renderização: Gislotica.

A câmara é o componente da máquina que determinará a posição absoluta da referência do pneu. Como se pode ver na figura 2.4, pode transladar na vertical e na horizontal através da montagem em dois eixos lineares servomotorizados. Sabe-se à partida que utiliza-se uma câmara da linha "Gocator" da marca LMI Technologies: a leitura de toda a superfície de um pneu requer que o próprio pneu seja rodado em torno do centro do transportador.

O LASER é verticalmente solidário à câmara, mas é possível movê-lo independentemente na direção horizontal. A empresa Alemã Koenig & Bauer faz parceria com a Gislotica, e foi imposto desde início que seria este o fornecedor deste equipamento. Este dispositivo não tem nenhuma restrição relativamente ao conteúdo da imagem a ser gravada e o cliente é quem decide o que quer gravar, dentro de uma área de 100 mm x 100 mm.

O transportador da máquina é fora do comum: contém um rasgo retangular no

seu centro, por onde pode transladar uma mesa. Para além de transladar na vertical, esta mesa também pode ser rodada em torno do seu centróide, permitindo colocar o pneu numa posição angular desejada. O acionamento vertical é feito através de um cilindro pneumático, e a rotação através de um servomotor.

Para garantir que o pneu é centrado no transportador, podem-se verificar dois centradores em forma de pescoço de cavalo. O seu acionamento linear é através de um servomotor e a uma transmissão correia-polia.

Quer para os transportadores das estações de entrada e saída, quer para o da estação central, serão utilizados rolos da marca Interrol. Estes rolos permitem a construção de um transportador pouco espaçoso porque contêm um motor e redutor embebidos. São componentes disponíveis em vários tamanhos e bastante utilizados nesta indústria. Acoplado a um dos rolos, irá estar um codificador angular para poder determinar a posição do pneu ao longo do transportador e medir o seu diâmetro externo.

Finalmente, é possível identificar na figura 2.4 quatro sensores fotoelétricos colocados ao longo do transportador. Aqueles que estão nos extremos são usados para detetar um pneu nas extremidades quando se está a receber ou a fornecer um pneu. A colocação de mais um sensor de cada lado dos portões, permite saber quando é ou não possível fechar o portão através da deteção de objetos de ambos os lados.

Para efetuar a medição da largura do pneu, utiliza-se uma barreira de medição que não está representada na figura 2.4. Esta barreira é colocada após o portão de entrada e antes do eixo de centragem. Desta forma, a presença de um objeto morfologicamente inesperado, será detetada antes de lhe começarem a ser realizadas operações.

Como já foi referido, trata-se de uma máquina para gravação de pneus do tipo PCR, o que significa que os pneus podem ter entre 440 mm e 800 mm de diâmetro externo e entre 165 mm e 350 mm de largura. Na prática, qualquer tipo de pneu pode ser processado, desde que as suas dimensões sejam compatíveis com a dimensão máxima dos acessos à máquina. Pneus de pequenas dimensões também não devem ser processados porque poderão ser incompatíveis com o curso dos eixos lineares.

Num painel adjacente à máquina é colocada a HMI, composta por um ecrã tátil e um conjunto de elementos de comando (várias botoneiras e um seletor). Estes elementos, as suas cores e as respetivas funções serão apresentados na secção 4.2.1 - Elementos de comando da HMI do capítulo 4. No ecrã tátil deverá ser apresentada a visualização o SCADA da RA, que por sua vez corre num computador industrial dentro do quadro elétrico.

Para finalizar esta secção faz-se referência a uma virtualização utilizada pela empresa para anunciar a máquina a possíveis clientes - figura 2.5. A observação desta imagem torna-se bastante útil para o leitor entender visualmente o aspeto da máquina. Salienta-se que nessa imagem estão representadas as estações de viragem do pneu em vez das estações de entrada e saída.

Na mesma figura, pode-se verificar que os componentes da estação central estão colocados dentro de uma estrutura metálica que os separa do exterior. Isto deve-se ao facto de a radiação emitida pelo LASER poder ser refletida na superfície do pneu e causar ferimentos ou dano, e a borracha libertar gases tóxicos ao derreter, que serão devidamente extraídos por um exaustor.

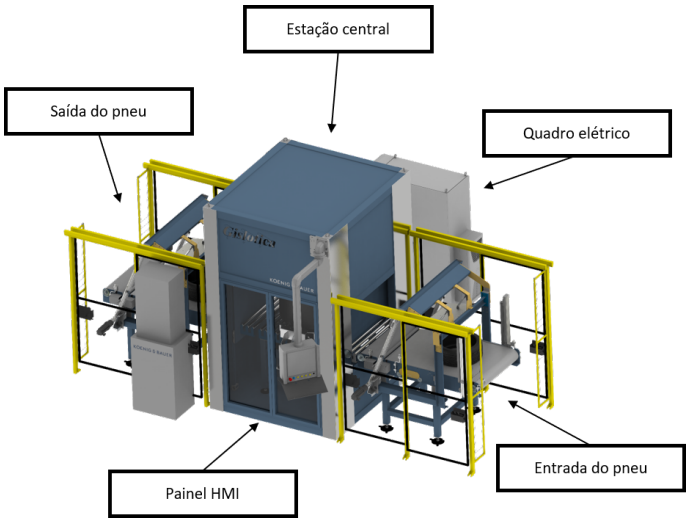


Figura 2.5: Renderização de uma possível configuração final da máquina. Renderização: Gislotica.

2.3 Funcionamento pretendido

Uma máquina industrial desta categoria deve apresentar dois modos de funcionamento para distinguir entre funcionamento autónomo e não autónomo. Nesta secção serão descritos ambos os modos de funcionamento, bem como o ciclo automático pretendido para realização do processo de gravação do pneu.

A colocação da máquina em cada modo deve ser executada pelo operador através de um seletor colocado no painel da HMI, que é denominado "seletor manual/automático". A posição manual corresponde ao modo não autónomo (ou modo manual), e a posição automático corresponde ao modo autónomo (ou modo automático).

No funcionamento autónomo, a máquina deve realizar o processo sem a interação do ser humano, desde que várias condições de segurança e processo sejam respeitadas. Dentro deste modo, deve ser planeado um ciclo de funcionamento em que a máquina realiza o processo de gravação, e outro na qual a máquina certifica que se pode entrar no ciclo de gravação, através da verificação de todas as condições necessárias para o processo se dar. Apesar de autónomo, o operador deve ter capacidade de interromper, suspender e retomar este ciclo.

Já no modo manual, deve ser possível configurar e testar a máquina, preparando-a para o modo automático. O operador deve poder realizar movimentos individuais de cada um dos componentes da máquina, permitindo verificar se estão a funcionar devidamente. Neste modo, o acesso à máquina deve ser permitido desde que todas as condições de segurança estejam reunidas.

As estações devem estar sempre todas no mesmo modo de funcionamento. Ou seja, não se prevê que seja possível colocar uma estação em modo manual enquanto as restantes estão em modo automático.

A sequência de ações realizadas durante o ciclo de gravação do pneu durante o modo automático deve ser prevista para que possa ser traduzido para o controlador da máquina. Esta análise é importante para expor o ciclo automático, em termos

de acionamento de componentes e obter uma primeira aproximação do tempo total de ciclo.

Na figura 2.6 está representado o diagrama de Gant do processo, feito de acordo com aquilo que foi descrito na secção 2.1 - Fluxo do processo de Marcação de um pneu. Neste diagrama, estão discretizadas as várias ações realizadas ao longo de um ciclo de funcionamento automático, bem como o tempo gasto em cada uma.

Os intervalos de tempo dedicados a cada tarefa foram determinados após discussão com a empresa, sempre utilizando as regras do "bom senso". Este é um assunto complicado de abordar porque se trata da primeira máquina para pneus deste tipo e são desconhecidos tempos necessários para realizar certas etapas, nomeadamente a etapa do LASER e da câmara. Assim, torna-se impossível à partida determinar quanto tempo demorará um ciclo no total.

O ciclo começa com a entrada de um novo pneu na estação central e termina com a saída do mesmo, após todas as etapas terem sido realizadas. Ao entrar, o pneu carece de uma centragem para garantir o sucesso da realização das etapas seguintes. Para tal o eixo de centragem deve avançar e recuar de forma a arrastar o pneu para a posição desejada, demorando 1 s em cada movimento.

De seguida, é acionado um cilindro pneumático, elevando a mesa rotativa a cerca de 60 mm acima do nível dos rolos do transportador, possibilitando o posicionamento angular da mesa em torno do seu próprio eixo. Este movimento deve ter uma duração de cerca de 2 s.

Nessa altura, uma câmara de visão é posicionada vertical e horizontalmente, dependendo da geometria do pneu, e procede à leitura da superfície do mesmo. Tanto o movimento vertical como o horizontal devem durar cerca de 2 s.

Durante a etapa da câmara, deve estar a ser executada uma rotação contínua do pneu. Este movimento é necessário para que a câmara possa percorrer toda a sua superfície, visto que a câmara não se move durante esta etapa.

Sabe-se que o tempo de duração desta etapa depende essencialmente da quantidade de pontos adquiridos da superfície do pneu. Por sua vez, a quantidade de pontos depende da resolução da câmara (distância lida no pneu entre dois feixes LASER), da sua frequência de aquisição (também chamada de *tick rate*) e do número de rotações executadas pelo pneu. Com base na experiência da empresa com pneus do tipo *Off The Road* (OTR), foi sugerida uma duração de 12 s para esta etapa.

Esta aproximação não é de todo precisa, porque um pneu do tipo PCR poderá requerer menor resolução e maior frequência de aquisição. No entanto, apenas é feita para obter uma noção da duração total do ciclo e, visto que o valor apresentado não influenciará nenhum outro cálculo importante, não haverá consequências graves.

Posteriormente, a mesa é reposicionada para a posição de gravação, etapa que demora 1 s a ser realizada. A câmara é recuada horizontalmente, e o LASER é posicionado vertical e horizontalmente para a gravação, demorando cada um destes três movimentos 2 s a ser executado.

Tecnicamente, seria possível ter as etapas de funcionamento do LASER e da câmara à mesma cota, mas não seriam condições ideais. Como isto não pode ser garantido à partida, deve-se considerar a possibilidade de ter de reposicionar o eixo vertical, tal como acabou de ser descrito.

Segue-se a etapa de gravação, cujo tempo demorado depende da imagem a gravar.

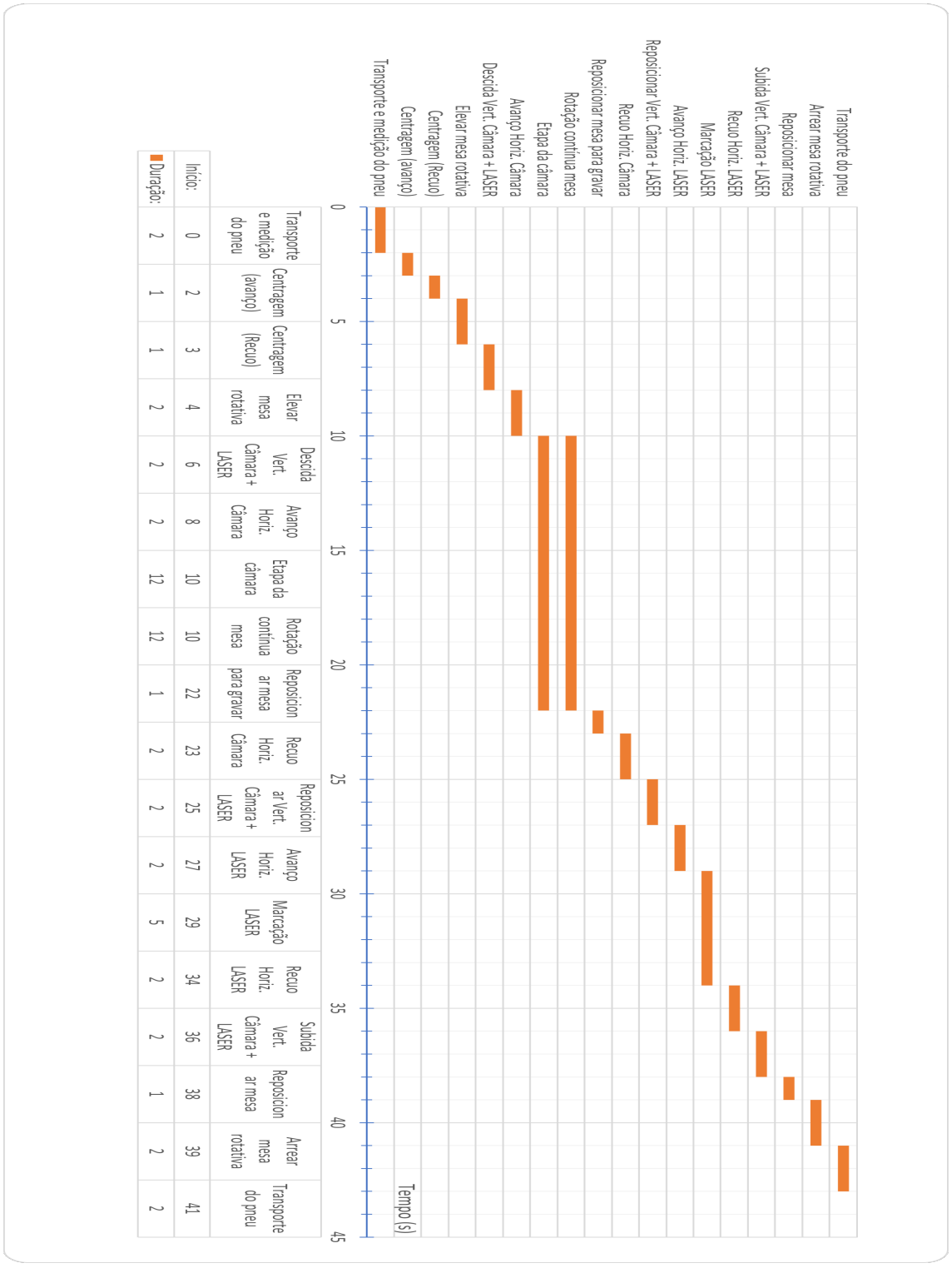


Figura 2.6: Diagrama de Gant do processo de gravação.

Como pode ser gravada qualquer imagem, decidiu-se que seriam dedicados 5 s a esta etapa, sabendo novamente que não é um valor exato.

Após a gravação estar concluída, o LASER é tornado à posição inicial através de um movimento horizontal e outro vertical, cada um com uma duração de 2 s. A mesa é colocada novamente na posição angular inicial e arreada, etapas que demoram respetivamente 1 s e 2 s.

Desta forma, o pneu é colocado de volta no transportador, para que possa ser transportado para fora da estação central, etapa que dura cerca de 2 s. Assim se

conclui um ciclo de funcionamento automático.

Após a máquina estar construída e o ciclo ser devidamente validado, então poder-se-ia proceder à movimentação contemporânea de certos eixos, para reduzir o tempo total de ciclo. Esta estratégia provocará trajetórias curvas difíceis de prever, pelo que é preciso especial cuidado neste aspeto.

Fica assim concluída a análise do ciclo de funcionamento automático, com uma duração total prevista de 43 s.

2.4 Considerações sobre a segurança da máquina

A segurança é um aspeto fundamental no desenvolvimento de qualquer máquina. Independentemente da zona do mundo para a qual a máquina será vendida e das normas aplicáveis, é imperativo que as situações perigosas sejam abordadas para que nunca ponham um operador em risco.

Apesar de não fazer parte dos objetivos da dissertação selecionar exatamente os componentes que se vão utilizar, é necessário prever o tipo de comportamento que se espera do sistema de segurança da máquina. Adicionalmente, será feita uma sugestão de componentes a incluir no sistema de segurança, que foi assumida como implementada aquando do desenvolvimento do *software* da máquina. A seleção dos componentes em si, de acordo com a norma ISO 13849-1 (que diz respeito a segurança de sistemas de controlo de máquinas), constitui assim um trabalho futuro.

Para poder saber em que aspetos se deve atuar, deve-se fazer uma análise de risco. Esta análise consiste na abordagem das várias situações perigosas uma a uma e tomar as medidas necessárias para que o operador esteja sempre em segurança. Estas medidas começam a ser implementadas durante a fase de projeto mecânico e prolongam-se ao longo de todo o desenvolvimento.

Caso os riscos não possam ser todos eliminados através de medidas de projeto mecânico, é preciso que seja implementado um sistema de comando de segurança, que é o caso. Para efetuar a seleção dos equipamentos do sistema de comando de segurança da máquina, devem-se analisar as várias situações perigosas que podem existir. Dos componentes que constituem a máquina, aqueles identificados como perigosos são:

- Eixos motorizados;
- Cilindros pneumáticos;
- Componentes que emitem radiação LASER;
- Transportadores das estações.

Como já foi explicado na secção 2.2 - Composição da máquina, a zona de trabalho da máquina é enclausurada numa estrutura metálica que impede qualquer acesso do exterior, delimitando a zona perigosa e impedindo o acesso do exterior. Desta forma, os riscos mecânicos (por exemplo esmagamento e corte) provocados pelos componentes móveis dentro dessa estação são eliminados, desde que o acesso do operador seja impossível em situações perigosas.

Adicionalmente, a solução construtiva dos transportadores das estações de entrada e saída é tal que não existem elementos móveis perigosos expostos (por exemplo engrenagens). Os únicos elementos móveis destas estações aos quais o operador tem acesso são os rolos dos próprios transportadores. Desta forma, os transportadores são componentes incapazes de provocar lesões graves.

Durante o ciclo, existem situações em que os portões abrem e fecham, criando uma situação possivelmente perigosa. Dentro desta situação, há três aspetos distintos a ter em conta, que serão analisados por ordem:

1. Acesso do operador ao interior da máquina quando o portão está aberto, podendo entrar em contacto com componentes em movimento;

2. Entalamento de partes do corpo de um operador entre o portão e a estrutura metálica da máquina;

3. Reflexão difusa da radiação LASER emitida para o exterior.

Para que não seja possível o operador alcançar a zona perigosa dos portões e muito menos o interior da máquina, devem ser colocados gradeamentos em torno de ambos os portões, visíveis na figura 2.7. Desta forma, os dois primeiros pontos ficam resolvidos porque a zona perigosa fica fisicamente delimitada e o risco é adequadamente reduzido.

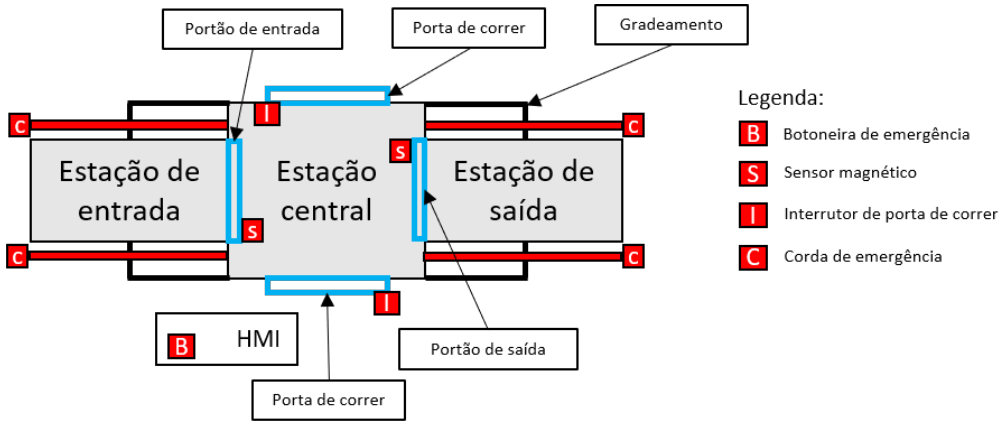


Figura 2.7: Disposição dos componentes de segurança da máquina.

Para dispositivos LASER de classe IV (que é o caso), a radiação refletida é perigosa porque pode causar queimaduras ou provocar incêndios [7]. Como tal, terá que se assegurar que este componente é inibido quando os portões não estão fechados através do sistema de comando de segurança.

A disposição planeada para garantir a segurança da máquina está representada na figura 2.7. Nesta figura, estão ainda representados os componentes do circuito de comando de segurança que se pretendem colocar, cuja justificação se apresenta nos parágrafos seguintes.

Decidiu-se implementar dois níveis de segurança hierárquicos. O primeiro nível corresponde a um nível de segurança global e o segundo a segurança parcial.

O primeiro, pode ser visto como um estado de emergência geral que faz com que todos os dispositivos perigosos sejam desprovidos de energia. Caso algum dispositivo esteja em movimento durante a queda da segurança geral, dá-se uma paragem o mais rápido possível. Neste nível, dá-se uma situação de emergência sempre que o espaço da estação central da máquina seja invadido, ou a pedido de um operador que detete uma situação perigosa.

O segundo nível deve inibir a emissão de radiação LASER sempre que algum dos portões não se encontrar fechado.

Para realizar as funções descritas, sugere-se a utilização dos seguintes dispositivos de segurança:

- Botoneira de emergência situada no painel de comandos do operador:

  Constitui um componente essencial que serve para provocar uma situação de emergência de nível um;

- Interrutores de segurança com trinco em cada uma das portas de correr:

  Servem para detetar uma abertura da porta de correr e impedir que esta seja aberta durante o ciclo. Caso não se utilizassem interrutores com trinco, o operador poderia ter tempo suficiente para abrir a porta e contactar com um eixo em movimento. A abertura de qualquer uma destas portas provoca a situação de emergência de nível um;

- Sensores magnéticos (sem contacto) de segurança, colocados de modo a detetar a posição "fechado´´ dos portões de entrada e saída:

  São utilizados para inibir o funcionamento do LASER quando o portão não está fechado, que corresponde a uma situação de emergência de nível dois;

- Uma corda de emergência colocada paralelamente ao lado dos transportadores:

  Nas fábricas com extensas linhas de transportadores é comum recorrer a este tipo de dispositivos. Desta forma, um operador que esteja posicionado ao lado do transportador também pode provocar uma situação de emergência de nível um.

Para implementar isto, por exemplo, poderiam ser colocados dois relés de segurança, um para cada nível. O primeiro relé serve para monitorizar o estado de emergência geral e a sua malha de segurança contém a botoneira de emergência, os interrutores de segurança e a corda de emergência. O segundo serve para monitorizar se ambos os portões estão fechados e a sua malha de emergência é composta por um sensor de segurança magnético sem contacto para cada portão.

A ligação elétrica deve ser feita de tal modo que uma queda da segurança no primeiro relé provoque a desenergização de todos os componentes perigosos, mas a queda da segurança do segundo relé apenas iniba a emissão de radiação LASER.

Em situações de ensaio ou manutenção da máquina estes níveis mantêm-se. Caso se pretenda aceder ao interior da estação central, todos os componentes perigosos deverão ser desprovidos de energia. Para que não exista a possibilidade de reenergização dos componentes enquanto alguém está a trabalhar na máquina, pode-se

recorrer a equipamentos para bloquear os componentes de segurança numa posição segura.

A última frase refere-se especialmente ao sistema de *Lockout/Tagout*, que constitui um procedimento de segurança utilizado em ambientes industriais para garantir que as máquinas estão devidamente desligadas e não começam a trabalhar antes da conclusão dos trabalhos de manutenção. Este sistema materializa-se, por exemplo, através de um cadeado que se coloca a impedir que a botoneira de emergência deixe de estar na posição atuada, que as portas de correr sejam fechadas ou que o seccionador geral do quadro elétrico saia da posição de corte de corrente.

Para continuar este trabalho segundo a ISO 13849-1, dever-se-ia começar por fazer uma determinação do *Performance Level* (PL) requerido da máquina através da análise de risco para cada uma das funções de segurança pretendidas. Depois, proceder à seleção dos devidos componentes de segurança (sensores, lógica e saídas) e calcular o PL atingido. Caso o PL atingido para cada função de segurança seja inferior ao PL requerido, então os componentes devem ser novamente selecionados até permitirem atingir um PL igual ou superior ao PLr.

As funções de segurança neste caso são as paragens de emergência de cada um dos componentes perigosos da máquina. Os valores de PL são calculados com base nas características de cada dispositivo de segurança e da solução implementada como um todo. Uma ferramenta bastante útil para este tipo de projeto é o SISTEMA, para o qual existem bibliotecas com os dados técnicos dos dispositivos de segurança de várias marcas.

Partindo do pressuposto de que existem pelo menos estes dois níveis de segurança, é feita a programação do *software* da máquina. Contudo, realça-se que alterações desta arquitetura são facilmente acomodáveis na lógica implementada, porque a segurança está num nível lógico independente. Deste modo, as escolhas tomadas aqui não têm impacto significativo naquilo que se apresenta nos próximos capítulos.

2.5 Síntese do capítulo

O capítulo que aqui termina deverá ter familiarizado o leitor com a máquina projetada pela empresa. Para tal, começou por se apresentar as etapas do processo a realizar, seguido da solução construtiva de projeto mecânico.

Na sequência, foram descritos os modos de funcionamento a implementar e ordem de movimento dos componentes ao longo de um ciclo do processo. Finalmente, foram apresentadas as considerações tomadas no que toca à segurança da máquina.

Após a leitura deste capítulo o leitor deve ter compreendido o contexto do dimensionamento e seleção de componentes e programação da máquina, questões a tratar nos capítulos que se seguem.

Capítulo 3

Seleção de componentes

Após o conhecimento dos elementos que constituem a máquina aqui tratada, inicia-se um novo capítulo relativo à seleção de componentes que se pode dividir em duas fases. A primeira diz respeito aos componentes do circuito de potência, que inclui servomotores e caixas redutoras, controladores (ou *drives*) dos motores e outros componentes do circuito elétrico. A segunda fase diz respeito à seleção de componentes do circuito de comando da máquina, nomeadamente PLC, computador industrial e sensores.

3.1 Componentes do circuito de potência

A empresa determinou que deveria ser feita a seleção de servomotores e *drives* para duas soluções distintas: Siemens e RA. A razão pela qual se tomou esta decisão foi dar ao autor oportunidade de se inteirar com as gamas disponibilizadas por ambas as marcas. Desta forma, se um futuro cliente decidir que quer optar por uma das duas marcas, a seleção destes componentes já estará feita.

Em primeiro lugar será feita a seleção de servomotores e caixas redutoras para os vários eixos da máquina. Seguidamente, passar-se-á à escolha dos drives. Como ferramentas para seleção de componentes, recorreu-se ao SIZER da Siemens e ao Motion Analyzer da RA, que são aplicações que facilitam a criação de perfis de movimento e disponibilizam os modelos de servomotores, *drives* e outros elementos dos circuitos de comando e potência da sua marca.

Visto que esta secção se tornaria demasiado extensa em termos de cálculos apresentados para os cinco eixos, decidiu-se utilizar a seguinte estratégia: será apresentada nesta secção a metodologia teoricamente seguida durante a seleção, e colocados no anexo A todos os cálculos que foram feitos para validar os resultados obtidos nas aplicações de dimensionamento.

3.1.1 Seleção de servomotores e caixas redutoras

Para os servomotores da RA[1], havia possibilidade de escolha entre duas séries de servomotores: MPL e VPL. Essencialmente, a diferença está no facto de os VPL

[1] O manual técnico destes servomotores está disponível em [8].

utilizarem apenas um cabo para o codificador, alimentação do servomotor e travão [9], enquanto os MPL utilizam um cabo para o codificador e outro para alimentação do servomotor e travão.

A vantagem dos VPL relativamente aos MPL vê-se no custo do cabo, pois o cabo único demora menos tempo a colocar e é mais barato [10–12]. Por outro lado, para a série MPL, a substituição apenas de um cabo de codificador não implica a substituição do cabo de alimentação do servomotor e travão. Dito isto, e na falta de outros requisitos (tais como preferência de um cliente), optou-se pela solução mais comum, escolhendo os MPL. Note-se que ambos os modelos (MPL e VPL) têm servomotores equivalentes, pelo que mais tarde pode-se alternar pelo equivalente da outra série.

Para os servomotores da Siemens, optou-se pela série 1FK7[2]. As alternativas seriam os da série 1FT7 (motores de alta performance) ou 1FK2 (motores com apenas um cabo para alimentação do motor, travão e sinal do codificador).

Para poder fazer a correta seleção e dimensionamento de um servomotor e de uma caixa redutora, é necessário fazer uma análise das cargas durante o ciclo de funcionamento nos respetivos eixos. O tempo de trabalho em ciclo de cada eixo considerado deveria ser resultado da análise do ciclo de funcionamento automático da máquina. Para efeitos de cálculo, no entanto, não se considerará o ciclo apresentado na secção 2.3 - Funcionamento pretendido do capítulo 2 através do diagrama de Gant da figura 2.6. As razões para não se ter utilizado este ciclo para efeitos de dimensionamento são as seguintes:

1. Estar-se-ia a negligenciar movimentações do eixo em modo de comando pelo operador (ou noutros ciclos da máquina não planeados), que poderiam levar a um sobreaquecimento do servomotor;

2. Não seriam permitidas alterações (imprevistas à partida) para reprogramação do ciclo de gravação;

3. Não se poderiam fazer *bypasses* de etapas, por exemplo no caso de o pneu ter de ser imediatamente expelido;

4. Os intervalos de tempo da etapas do ciclo ainda não são bem conhecidos.

Alternativamente, será considerado um ciclo específico para cada eixo, descrito na respetiva secção do anexo A, com um *Duty Cycle* (DC) de 50 %. Todas estas decisões foram sustentadas pela empresa.

Nesta máquina existirão cinco servomotores (que podem ser visualizados na figura 2.4), dentro dos quais dois são idênticos. Para facilitar a referência a cada um dos motores, serão codificados daqui em diante pelas letras A a D, e serão apresentados pela ordem abaixo:

1. Servomotor A : Acionamento do movimento linear que centra o pneu no transportador;

[2]O manual técnico destes servomotores está disponível em [13].

2. Servomotor B : Acionamento do movimento linear vertical do LASER e da câmara;

3. Servomotor C : Acionamento do movimento linear horizontal do LASER ou da câmara - propositadamente dois servomotores iguais;

4. Servomotor D : Acionamento do movimento angular da mesa rotativa.

Adicionalmente, qualquer referência ao eixo A corresponderá a uma referência ao eixo acionado pelo servomotor A, e o mesmo para os restantes eixos.

A metodologia seguida durante a seleção será explicada através de uma introdução teórica onde se apresentarão as fórmulas utilizadas de maneira generalizada para poder corresponder ao dimensionamento de servomotores e respetivas caixas redutoras para todos os eixos presentes.

Considere-se o movimento linear de um ponto até ao outro, de uma carga, acionada por um conjunto correia - polia, como o da figura 3.1.

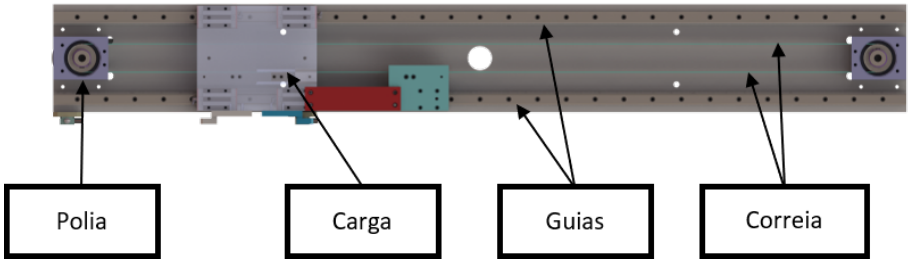


Figura 3.1: Sistema de acionamento correia-polia. Projeto: Gislotica.

Considere-se que o perfil de velocidade desse movimento tem a forma trapezoidal (figura 3.2), sendo constituído por uma fase de aceleração, uma fase em movimento linear uniforme e uma desaceleração até à paragem. A distância percorrida - d_{perc} pode ser calculada em função da velocidade durante o movimento uniforme - v_{const}, do tempo total de avanço - $t_{avanço}$, do instante de tempo em que termina a aceleração - t_{acel} e do instante de tempo em que começa a desaceleração - $t_{desacel}$[3], através da equação:

$$d_{perc} = \int_{0}^{t_{acel}} \left(\frac{v_{const}}{\Delta t_{acel}} \times t\right)\, dt + v_{const} \times (t_{desacel} - t_{acel}) + \int_{t_{avanço}-t_{desacel}}^{t_{avanço}} \left(v_{const} - \frac{v_{const}}{\Delta t_{acel}} \times t\right)\, dt \tag{3.1}$$

A aceleração inicial é dada pela equação 3.2.:

$$a_{inicial} = \frac{v_{const}}{\Delta t_{acel}} \tag{3.2}$$

Sabendo a massa da carga - m_{carga}, a aceleração - $a_{cel\text{-}carga}$, a força de atrito - F_{atrito} e o rendimento do conjunto correia-polia (incluindo guias) - $\eta_{c\text{-}p}$, é possível calcular

[3]Em todos os cálculos será considerado um intervalo de tempo de aceleração igual ao de desaceleração. Desta forma, $\Delta t_{acel} = \Delta t_{desacel} = t_{acel}$.

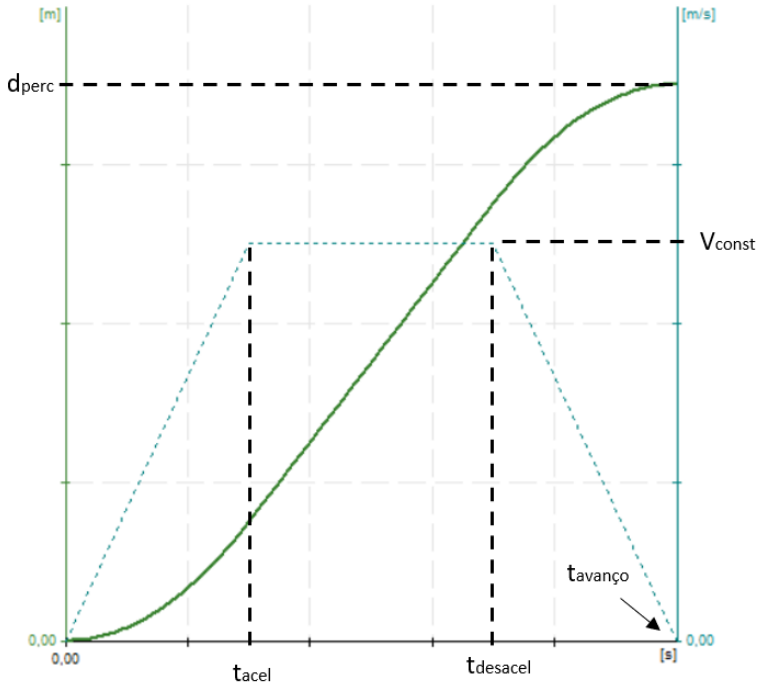


Figura 3.2: Perfil temporal de posição (traço contínuo verde) e velocidade (traço interrompido azul) a considerar.

a força necessária para mover a carga - F_{carga}. No entanto, a equação utilizada para o seu cálculo depende da situação.

Caso o servomotor esteja a aplicar binário no sentido de movimento da carga, é considerado uma fonte de energia - um motor. O atrito (sob a forma de rendimento) está a contribuir para uma maior necessidade de energia fornecida pelo servomotor.

Por outro lado, quando o servomotor está a aplicar binário no sentido contrário ao do movimento da carga, o servomotor estará a consumir a energia do movimento. Neste caso, diz-se que está a funcionar como gerador, e o atrito (sob a forma de rendimento) contribui para uma menor quantidade de energia gerada.

Na equação 3.3 estão representadas as equações utilizadas para ambas as situações. Será desprezada a massa da correia e inércia da polia porque são grandezas com pouco peso no resultado final. Adicionalmente, o facto de não se considerar um coeficiente de atrito das guias é compensado sob a forma de rendimento do conjunto.

$$\begin{aligned} F_{carga\ -\ motor} &= (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \frac{1}{\eta_{c\text{-}p}} \\ F_{carga\ -\ gerador} &= (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \eta_{c\text{-}p} \end{aligned} \tag{3.3}$$

Dado o diâmetro da polia - ϕ_{polia}, é possível calcular o binário que deve ser exercido à entrada da polia - T_{polia}, através da equação 3.4.

$$T_{polia} = F_{carga} \times \frac{\phi_{polia}}{2} \tag{3.4}$$

A velocidade angular da polia- ω_{polia} é dada por:

$$\omega_{polia} = \frac{v_{carga}}{\frac{\phi_{polia}}{2}} \tag{3.5}$$

No caso do acionamento de uma carga angular logo à saída da caixa redutora, o binário e velocidade da carga correspondem às variáveis T_{polia} e ω_{polia}, respetivamente. Esta informação será útil aquando do dimensionamento dos componentes para o eixo D.

Sabendo a razão de transmissão da caixa redutora - $i_{redutora}$, o seu momento de inércia visto pelo servomotor - $J_{redutora}$ e o seu rendimento - $\eta_{redutora}$, o binário à entrada da caixa redutora - $T_{redutora}$ é dado pela equação 3.6:

$$\begin{aligned} T_{redutora\ -\ motor} &= (\frac{T_{polia}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{servo}}{\Delta t_{acel}}) \times \frac{1}{\eta_{redutora}} \\ T_{redutora\ -\ gerador} &= (\frac{T_{polia}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{servo}}{\Delta t_{acel}}) \times \eta_{redutora} \end{aligned} \tag{3.6}$$

E a velocidade angular do servomotor ω_{servo} dada pela equação 3.7:

$$\omega_{servo} = \omega_{polia} \times i_{redutora} \tag{3.7}$$

Por fim, sabendo o momento de inércia do veio do servomotor - J_{servo}, é possível calcular o binário à saída do servomotor - T_{servo} através da equação 3.8:

$$T_{servo} = T_{redutora} + J_{servo} \times \frac{\omega_{servo}}{\Delta t_{acel}} \tag{3.8}$$

Nesta etapa do dimensionamento seria necessário selecionar uma caixa redutora da marca Wittenstein Alpha (exceto para o eixo D, para o qual a empresa já havia selecionado uma caixa redutora da marca SEW). Esta marca apresenta uma ampla gama de servoredutores, e no catálogo consultado [14] especifica que a gama de melhor relação qualidade-preço para aplicações com tolerância de posicionamento média é a gama *Alpha Value Line*.

Dentro desta gama, foram escolhidas caixas planetárias porque estão disponíveis numa grande gama de razões de transmissão e de binário máximo, com um *design* compacto. Como a saída da redutora não é para acoplar diretamente na polia, e não necessita de nenhum tipo de furação específica, escolheu-se a série NP. Esta série tem ainda a vantagem de ser da gama de baixo *backlash*, garantindo um menor erro no posicionamento, menor ruído e menor desgaste.

O *backlash* consiste na folga entre as involutas de dois dentes de uma engranagem, que contactam quando se inverte o sentido de rotação. Esta folga é inevitável e leva a um aumento do erro de posicionamento.

O correto dimensionamento de uma redutora traz várias vantagens, tais como:

- Diminuir o binário requerido no servomotor, permitindo escolher servomotores de menor dimensão;
- Colocar o motor a funcionar a rotações mais altas, onde o seu funcionamento tem maior rendimento;

- Um aumento da razão de transmissão contribui de forma quadrática para a redução do valor da razão de inércias.

Assim, durante o dimensionamento tentou-se colocar uma caixa redutora com razão de transmissão o mais alta possível, de modo a não exceder os valores limite de funcionamento tanto da caixa como do servomotor.

Para proceder à seleção de uma caixa redutora, é importante ter especialmente em conta as seguintes grandezas:

- O binário máximo: nunca deve ser excedido, com um coeficiente de segurança $N = 1,5$;
- O binário de aceleração máximo (menor que o binário máximo) : não deve ser excedido;
- A velocidade média de entrada: não deve ser excedida[4];
- A razão de transmissão: quanto maior, menor será o esforço no servomotor, e maior a sua velocidade de rotação;
- O diâmetro do cubo de entrada: deve ser compatível com o servomotor selecionado.

Após a seleção da redutora, seleciona-se um servomotor que se adeqúe. Desta forma, termina-se com o motor e a redutora dimensionados à medida, ou seja, com o esforço distribuído pelos dois componentes. Para o correto dimensionamento de um servomotor, serão verificados os seguintes aspetos:

- Se os todos pontos de funcionamento estão dentro da curva de binário do servomotor;
- Se o binário em qualquer ponto do ciclo não excede 75 % do valor máximo permitido;
- Se o binário médio efetivo calculado não excede 75 % do valor da curva de regime permanente, à velocidade média;
- Se a razão de inércias entre o momento de inércia reduzido da carga e do servomotor não excede um determinado valor;
- Se o motor é o mais ajustado à situação de entre os restantes disponíveis.

De acordo com a empresa, não são incomuns casos em que um servomotor de maior capacidade tem um preço mais competitivo do que um servomotor mais pequeno. No decorrer desta seleção não houve atenção a preços de mercado, porque o objetivo era seguir as regras da teoria.

O binário médio efetivo é uma medida da utilização térmica do servomotor. O valor que se calcula, é o valor do binário que se obteria caso fosse despendida a mesma energia que para o ciclo em questão, mas em regime de binário constante. É

[4] A velocidade média apresentada no catálogo não tem em conta instantes a velocidade zero.

seguido o mesmo princípio que para o cálculo da corrente eficaz num circuito elétrico de corrente alternada.

A utilização da média aritmética para obter este valor não faria sentido porque considera que valores positivos e negativos têm pesos opostos, cancelando-se. Adicionalmente, também não se pode utilizar a média dos valores absolutos dos binários porque a potência total produzida não é linearmente dependente do binário.

Como tal, o cálculo do binário médio efetivo tem como finalidade garantir que o servomotor está abaixo de uma certa temperatura, para a qual se sabe que não há danos nos seus componentes, principalmente em pontos críticos como os ímans permanentes que podem sofrer deterioração das suas propriedades magnéticas e nos enrolamentos, cujo isolamento pode derreter e conduzir a um curto-circuito. Para proceder ao seu cálculo, utiliza-se a equação 3.9:

$$T_{eff} = \sqrt{\frac{\sum_{i=1}^{n} \Delta t_i \times {T_i}^2}{\sum_{i=1}^{n} \Delta t_i}} \tag{3.9}$$

A razão de inércias - $r_{inércias}$ é dada pela razão entre a inércia da carga vista pelo servomotor e a inércia do veio do servomotor. A inércia da carga vista pelo servomotor é obtida recorrendo a uma igualdade de energias cinéticas; procura-se um momento de inércia equivalente ao da carga - J_{eq}, caso não houvesse transmissão nenhuma:

$$\frac{1}{2} \times J_{eq} \times {\omega_{servo}}^2 = \frac{1}{2} \times m_{carga} \times {v_{carga}}^2 + \frac{1}{2} \times J_{carga} \times {\omega_{carga}}^2 \tag{3.10}$$

$$r_{inércias} = \frac{J_{eq}}{J_{servo}} \tag{3.11}$$

Os valores aceitáveis para este parâmetro são normalmente estabelecidos pelo fabricante para conseguir um bom funcionamento do sistema de controlo, e para servomotores são normalmente inferiores a 10 [13]. Um valor baixo de razão de inércias seria o ideal em termos de controlo , mas não é sempre exequível, porque significa o que servomotor está sobredimensionado, e há um gasto excessivo de energia e custos.

Para seleção de servomotores, utilizou-se a seguinte escala para atender à aceitabilidade da razão de inércias de um servomotor:

- De 1 a 3 - valores ótimos;
- De 3 a 7 - valores aceitáveis;
- De 7 a 10 - para aplicações pouco dinâmicas;
- Maior que 10 - valores não recomendados.

Para valores de razão de inércias acima do valor recomendado, o controlo de movimento deixa de ser preciso na medida em que podem surgir ressonâncias e anti-ressonâncias dentro da gama de funcionamento do servomotor. Isto quer dizer que para certas frequências de acionamento, o sistema vai apresentar uma resposta

inesperada e de amplitudes ressonantes, que afetará a sua capacidade de posicionamento. Isto pode traduzir-se em tempos de resposta elevados, sobrecursos (ou em inglês *overshoot*) ou, no pior dos casos, uma instabilidade no sistema sob forma de uma oscilação em torno de um ponto [15].

Outros aspetos importantes na seleção de um servomotor são o tipo de codificador, o prazo de entrega, a inclusão ou não de um travão e o tipo de veio de saída. Quanto aos codificadores, normalmente são disponiblizadas várias opções no que toca à sua resolução, multi-volta ou uni-volta: os uni-volta são normalmente utilizados em aplicações onde o servomotor não dá mais que uma rotação completa, que não é o caso, pelo que aqui se opta sempre por multi-volta. A resolução escolhida depende dos requisitos da aplicação e, na falta de requisitos, escolheu-se a maior disponível; contudo, na maior parte dos casos aqui abordados apenas estava disponível um valor de resolução.

No decorrer do dimensionamento de servomotores, não houve preferência por motores com curtos prazos de entrega. Escolheram-se sempre servomotores com travão, porque é uma prática padrão na empresa, salvo casos pontuais. O tipo de veio dos servomotores escolhidos é liso em todos os eixos, exceto no eixo D que deve ser escatelado para ter a devida interface com a caixa redutora.

A tensão nominal de alimentação dos servomotores escolhidos é 400 V AC. Caso seja necessário consultar as listas de servomotores disponíveis da Siemens ou da RA estas encontram-se em [16] e [17], respetivamente.

Após esta introdução, poder-se-ia proceder aos cálculos individuais para cada eixo. Repete-se que todos os cálculos e passos realizados, análise detalhada dos eixos e os modelos de servomotores e caixas redutoras escolhidos estarão disponíveis anexo A - Cálculos realizados para a seleção de servomotores e caixas redutoras.

Tabela 3.1: Servomotores e caixas redutoras escolhidos para as soluções Siemens e RA.

Eixo	Servomotor	Caixa redutora	Utilização térmica (%)	Binário máximo (%)	Razão de inércias
Solução Siemens					
A	1FK7022-5AK7	NP015S-MF2-20	52	43	5,89
B	1FK7061-4CF7	NP035S-MF2-16	60	40	1,38
C	1FK7011-5AK7	NP005S-MF2-20	30	51	6,95
D	1FK7042-2AF7	KH47/R-1400/29	56	68	4,55
Solução Rockwell Automation					
A	MPL-B1530U	NP015S-MF2-20	41	44	5,8
B	MPL-B430P	NP035S-MF2-16	74	69	1,67
C	MPL-B1510V	NP005S-MF2-20	32	36	7,5
D	MPL-B420P	KH47/R-1400/29	38	62	4,83

Os modelos de servomotor e as caixas redutoras escolhidos para cada eixo são apresentados na tabela 3.1. Nas colunas são apresentadas as métricas mais importantes para a seleção. A utilização térmica[5] e o binário máximo são dados em

[5]É a razão entre o binário efetivo e o ponto da curva de regime permanente do motor à velocidade

percentagem da razão entre o valor máximo calculado no ciclo e o valor máximo permitido de acordo com o fabricante e permitem entender que os valores destas métricas para os conjuntos servomotor/caixa redutora selecionados estão dentro dos intervalos especificados.

3.1.2 Seleção de componentes elétricos do circuito de potência

Nesta secção são dimensionados os componentes elétricos de potência para acionamento dos servomotores já escolhidos. A seleção destes componentes também foi feita com recurso às aplicações de apoio ao dimensionamento de ambas as marcas. Os componentes a considerar nesta secção são:

- *Drives* de servomotores;
- Filtro de linha;
- Reator de linha;
- Resistência de frenagem.

Como esta análise não será tão extensa como a feita para a secção anterior, os componentes disponíveis abordados de cada marca serão apresentados no corpo da dissertação.

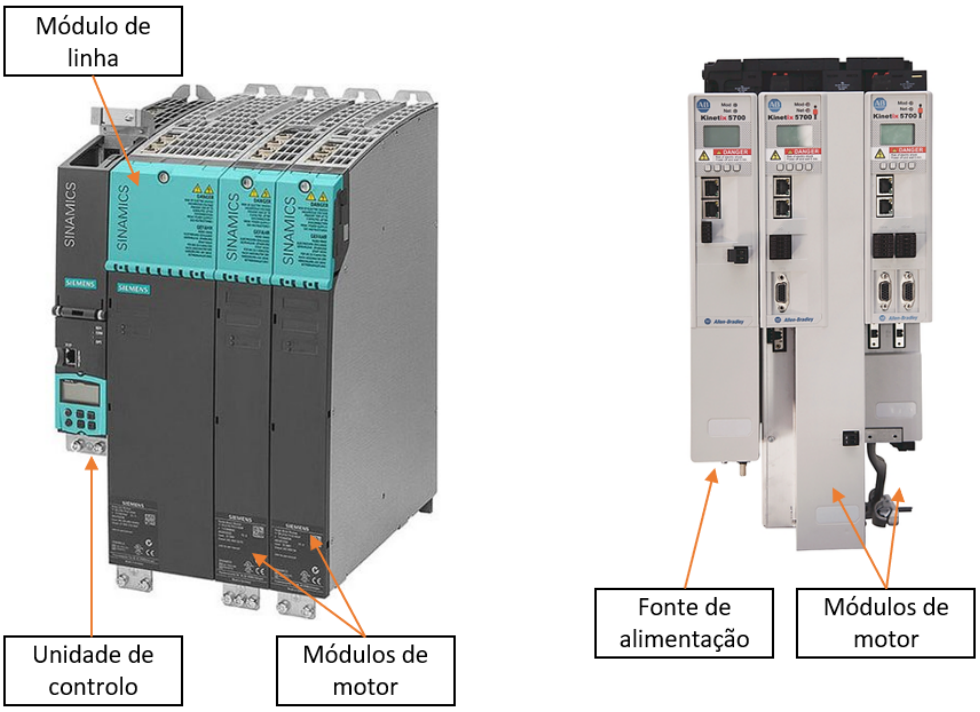
Independentemente da marca utilizada, visto que existem pelo menos cinco servomotores planeados nesta máquina, opta-se pela utilização de *drives* de servomotores com uma fonte de alimentação comum. Alternativamente, poder-se-iam utilizar *drives* com fonte de alimentação individual para cada motor. A vantagem dos escolhidos é tornarem-se mais económicos à medida que aumenta o número de servomotores presente porque reduzem a quantidade de cabos passados e ocupam menos espaço no quadro elétrico.

A arquitetura das unidades escolhidas é semelhante em ambas as marcas, mas não é igual. Para a Siemens, são necessários três componentes:

1. Unidade de controlo - Responsável por fazer o controlo do movimento do servomotor;
2. Módulo de linha - Responsável pela retificação da alimentação trifásica e criação de uma tensão contínua no barramento DC;
3. Módulo de motor - Responsável pela inversão da tensão presente no barramento DC para alimentação AC dos enrolamentos do estator do servomotor, em função das ordens de comando.

No caso da RA, a unidade de controlo não existe como um componente separado e o controlo de movimento é feito por parte do PLC, que necessita de ser um PLC dito de *motion*. Adicionalmente, esta marca designa o módulo de linha por fonte de alimentação. Na figura 3.3 estão representadas as famílias que serão abordadas de ambas as marcas.

média do ciclo.



(a) Siemens Sinamics S120. (b) Rockwell Automation Kinetix 5700.

Figura 3.3: *Drives* de motores considerados de ambas as marcas.

Para situações de emergência, os módulos de motor contêm normalmente um conjunto de entradas físicas que correspondem a funções de segurança (figura 3.4). Estas funções podem incluir *Safe Torque Off* (STO), *Safe Stop 1* (SS1) , *Safe Stop 2* (SS2) , entre outas cuja descrição detalhada pode ser consultada em [16]. Os 24 V seguros saídos de um possível relé de segurança seriam ligados a estas entradas em questão, permitindo efetuar cada uma das funções de segurança.

Como esta aplicação não é temporalmente crítica, opta-se por apenas considerar necessário o STO. Com efeito, quando há uma situação de emergência, deverá ser executada uma ordem de paragem do servomotor. Independentemente de se dar ou não realmente a paragem, após um determinado intervalo de tempo deverá ser acionado o travão do servomotor e o STO, inibindo a geração de binário e levando-o a uma situação de velocidade nula. A ordem de paragem do servomotor não é uma ordem de segurança, mas a entrada STO é realmente um elemento de segurança de máquinas.

Consequentemente, não serão utilizados módulos de motor com funcionalidades de segurança, para além do STO.

O módulo de linha e os módulos dos motores estão ligados entre si por um barramento (literalmente barras metálicas) onde existe uma diferença de potencial entre 510 e 720 V DC. Este barramento é chamado de *DC link* e contém um condensador ligado entre o polo positivo e o negativo, que serve de acumulador de energia e também filtra transientes de tensão saídos da fonte de alimentação (tais como ruído

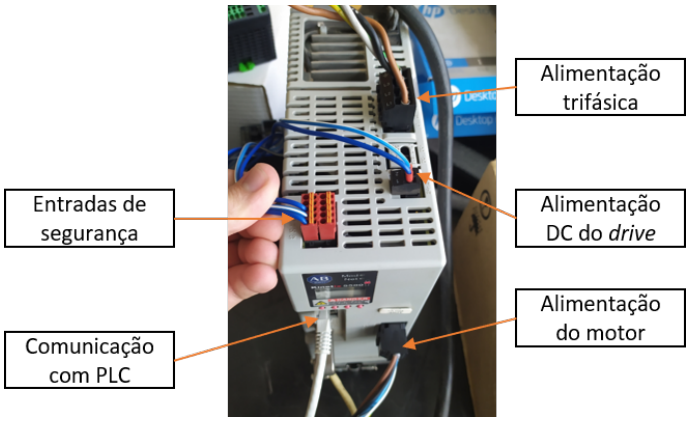


Figura 3.4: Detalhe das entradas de segurança (STO) de um *drive* de RA.

e picos).

O dimensionamento dos módulos é feito com base na corrente necessária no servomotor em regime permanente e de pico. Visto que o *software* faz imediatamente o cálculo da corrente necessária em cada servomotor, basta o utilizador escolher os módulos de cada motor que necessita e conjugá-los entre módulos duplos ou simples. Um módulo duplo alimenta dois motores ao mesmo tempo, ao passo que um módulo simples apenas alimenta um[6].

O filtro e o reator de linha são dois componentes que trabalham em conjunto para limitar a interferência introduzida pelo módulo de linha na rede [16]. O filtro reduz a amplitude dos harmónicos de maior ordem que possam influenciar negativamente outros equipamentos na rede elétrica para valores limite segundo a norma EN 61800-3[7] e é um componente obrigatório para que possa ser aposta a marcação CE numa máquina. Adicionalmente, o reator de linha serve para estabilizar a amplitude da tensão de alimentação, evitando picos que possam danificar os semicondutores dos módulos de linha e é um componente opcional.

A resistência de frenagem é um componente necessário para transformar a energia cinética acumulada num eixo em calor, à medida que o servomotor desacelera uma carga e para descarregar o condensador do barramento DC, em caso de necessidade. Quando a tensão no barramento excede um determinado valor, é permitida a passagem de corrente pela resistência de frenagem [18], que deverá estar colocada no exterior do armário. Na Siemens, é ligado ao barramento DC um módulo de frenagem que controla a passagem de corrente; na RA a resistência é ligada à fonte de alimentação.

Uma resistência de frenagem pode ser definida pela sua resistência elétrica e pela potência média que é capaz de dissipar. A resistência elétrica escolhida irá determinar a potência máxima que se consegue dissipar: quanto maior o valor da resistência, menor será a potência máxima. A resistência média é calculada com base na análise das necessidades de cada aplicação.

Para calcular a potência máxima, deve-se começar por identificar o eixo que

[6] A diferença prática reside no facto de um módulo duplo ser mais barato e ocupar menos espaço no quadro do que dois módulos simples.

[7] A norma EN 61800-3 contempla requisitos de compatibilidade eletromagnética e métodos para testar a interferência criada por componentes elétricos de acionamento.

consegue adquirir maior energia mecânica, calculada através das equações 3.12, 3.13 e 3.14. Nessas equações E_{mec} representa a energia mecânica, E_{pot} a energia potencial, E_{cin} a energia cinética, m é a massa da carga, g é a aceleração da gravidade, h é a altura percorrida durante a frenagem, v e ω são as velocidades linear e angular da carga, e J é o momento de inércia da carga.

$$E_{mec} = E_{pot} + E_{cin} \tag{3.12}$$

$$E_{pot} = m \times g \times h \tag{3.13}$$

$$E_{cin} = \frac{1}{2} \times m \times v^2 + \frac{1}{2} \times J \times \omega^2 \tag{3.14}$$

Seguidamente, seria necessário conhecer o tempo de frenagem pretendido numa situação de emergência e dividir a energia pelo intervalo de tempo, obtendo a potência. Esta aproximação é válida desde que a potência resultante seja inferior à potência máxima que o servomotor consegue desenvolver, ou posto de outra forma, o binário aplicado seja inferior ao binário máximo para cada velocidade[8]. Neste caso, o intervalo de tempo considerado para uma paragem de emergência será $\Delta t_{emg} =$ 50 ms.

Considere-se o eixo B a transladar de cima para baixo à velocidade máxima (v_{const} = 0,4 m/s) no qual se pretende realizar uma paragem de emergência. A energia mecânica que deve ser retirada, de modo a imobilizá-lo é calculada através das equações 3.12, 3.13 e 3.14:

$$\begin{aligned}
a_{cel} &= \frac{v_{const}}{\Delta t_{emg}} = \frac{0,4}{0,05} = 8\ m/s^2 \\
h &= v_0 \times t - \frac{1}{2} \times a_{cel} \times t^2 \\
h &= 0,4 \times 0,05 - \frac{1}{2} \times 8 \times 0,05^2 \\
h &= 0,01\ m \\
E_{pot} &= 200 \times 9,81 \times 0,01 = 19,62\ J \\
E_{cin} &= \frac{1}{2} \times 200 \times 0,4^2 = 16\ J \\
E_{mec} &= 19,62 + 16 = 35,62\ J \\
Pot_{max} &= \frac{35,62}{0,05} = 712,4\ W
\end{aligned}$$

Da mesma maneira, considere-se o eixo D a rodar a uma velocidade de 2π rad/s. Novamente, a energia mecânica que deve ser retirada, de modo a imobilizá-lo é calculada através das equações 3.12, 3.13 e 3.14:

[8]Se a potência calculada for maior do que aquela que o servomotor consegue desenvolver, errar-se-á por sobredimensionamento da resistência de frenagem, o que não é problemático.

$$E_{pot} = 0\ J$$
$$E_{cin} = \frac{1}{2} \times 3,3 \times (2\pi)^2 = 65,14\ J$$
$$E_{mec} = 0 + 65,14 = 65,14\ J$$
$$Pot_{max} = \frac{65,14}{0,05} = 1302,8\ W$$

Pelo que se determina que é o eixo D que necessitará do maior de potência durante uma paragem de emergência. Considerando uma tensão no barramento DC de 600 V, pode-se calcular a resistência máxima permitida para que seja possível dissipar esta potência a este valor de tensão[9]. Sabendo a potência de frenagem necessária - P_{ot} e a tensão no barramento DC - V_{DC}, a resistência - $\Omega_{res-frenagem}$ é dada pela equação 3.15:

$$\Omega_{res-frenagem} <= \frac{V_{DC}^2}{P_{ot}} \tag{3.15}$$

$$\Omega_{res-frenagem} <= \frac{600^2}{1302,8}$$
$$\Omega_{res-frenagem} <= 276\ \Omega$$

De seguida, deve-se calcular a potência média necessária para a frenagem durante um ciclo. Tecnicamente, dever-se-ia fazer uma análise tendo em conta o ciclo global para obter a potência média dissipada na resistência. Porém, na incerteza quanto à verdadeira duração do ciclo global e na necessidade de escolher uma resistência, faz-se o seguinte: analisa-se somente o ciclo utilizado para dimensionar o servomotor B que é o mais intenso em termos de potência.

Como se verá, os cálculos permitem concluir que a resistência de frenagem de menor capacidade disponível é suficiente para dissipar toda a energia necessária. Alguém mais experiente teria facilmente feito esta decisão, mas para efeitos de dissertação opta-se por incluir estes cálculos nos seguintes parágrafos.

A potência ao longo do ciclo do eixo B em função do tempo está representada na figura 3.5. Note-se que para o cálculo da potência dissipada será desprezada a inércia do motor e da redutora porque têm peso mínimo no cálculo final. Adicionalmente, o rendimento do mecanismo e do módulo do motor serão considerados 100 %.

Os vários níveis de potência na carga são calculados utilizando a fórmula $P_{carga} = T_{carga} \times n_{carga} \times \frac{2\pi}{60}$. Os binários e as velocidades ao longo do ciclo já foram previamente calculados na subsecção relativa ao dimensionamento do eixo B no anexo A.

[9]Caso se colocasse uma resistência maior do que a permitida, não seria possível obter a potência de frenagem necessária, aumentando o tempo necessário para imobilizar um eixo.

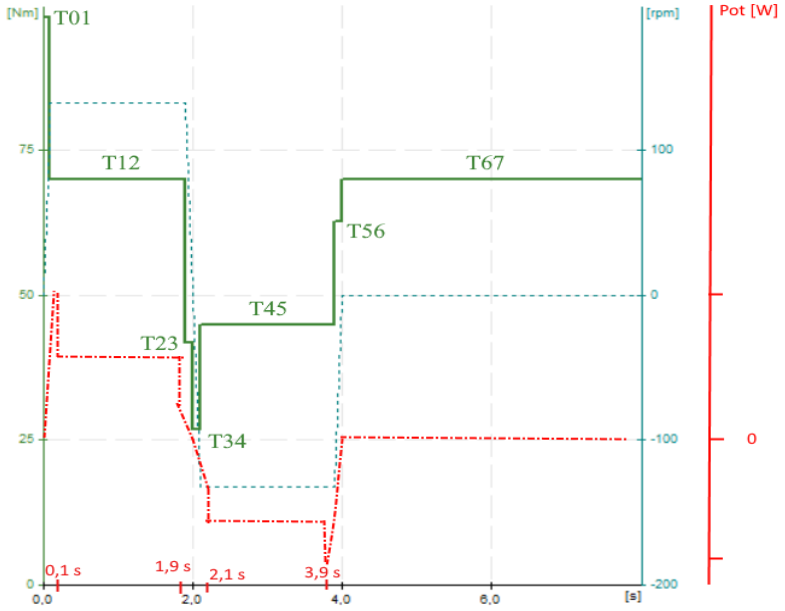


Figura 3.5: Perfil temporal do binário - T_{carga} (traço contínuo) e velocidade - n_{carga} (traço interrompido) e potência - P_{carga} (traço - ponto) da carga do eixo B. Fonte: SIZER da Siemens, editado.

$$P_{carga@t=0-0,1s} = 98,0 \times 132,26 \times \frac{2\pi}{60} = 1357\ W$$
$$P_{carga@t=0,1-1,9s} = 69,9 \times 132,26 \times \frac{2\pi}{60} = 968\ W$$
$$P_{carga@t=1,9s-2s} = 41,8 \times 132,26 \times \frac{2\pi}{60} = 579\ W$$
$$P_{carga@t=2s-2,1s} = 26,7 \times (-132,26) \times \frac{2\pi}{60} = -370\ W$$
$$P_{carga@t=2,1s-3,9s} = 44,7 \times (-132,26) \times \frac{2\pi}{60} = -620\ W$$
$$P_{carga@t=3,9s-4s} = 62,7 \times (-132,26) \times \frac{2\pi}{60} = -868\ W$$

A potência média dissipada $P_{diss-media}$ é dada pela média aritmética dos intervalos em que o motor funciona como gerador:

$$P_{diss-media} = \frac{P_{carga@t=2s-2,1s} \times 0,5 \times (2,1-2) + P_{carga@t=2,1s-3,9s} \times (3,9-2,1) + \frac{+P_{carga@t=3,9s-4s} \times 0,5 \times (4-3,9)}{0,1+1,8+0,1+}}{+0,1+1,8+0,1+4}$$

$$P_{diss-media} = \frac{-370 \times 0,5 \times 0,1 - 620 \times 1,8 - 868 \times 0,5 \times 0,1}{8}$$
$$P_{diss-media} = -147W$$

Desta forma, conclui-se que a potência média dissipada é 147 W. Este valor será retomado aquando da escolha de uma resistência de frenagem para ambas as marcas.

Feita esta análise incial, passa-se à seleção de componentes começando pela solução Siemens, cujo catálogo consultado está referenciado em [16]. Para a RA, o catálogo consultado está referenciado em [19].

Componentes do circuito elétrico de potência - Siemens

A Siemens disponibiliza módulos de motor em vários formatos: *Blocksize*, *Booksize* e *Chassis*. Nesta aplicação serão utilizados os módulos no formato *Booksize* (figura 3.3a) porque permitem ligação de vários módulos de motor em série, com uma fonte de alimentação e unidade de controlo comum.

Dentro do formato Booksize, existe a opção de escolher a versão compacta, denominada "Booksize Compact". Esta versão permite reduzir o espaço ocupado no quadro elétrico em troca de menor capacidade de sobrecarga[10], mas mantendo os valores de corrente nominal e limite da versão normal. Isto significa que apesar de ambos os módulos atingirem o equilíbrio térmico à corrente nominal, os compactos são mais rápido a atingi-lo.

A diferença entre módulos apenas é notória para ciclos com situações de corrente acima da nominal durante mais de 60 s [16], devido a uma menor inércia térmica. Nesta aplicação, como se pode ver pelos perfis temporais dos vários eixos disponíveis no anexo A, não há nenhum ciclo em que se esteja a trabalhar acima do binário efetivo (que corresponderá a uma corrente menor que a nominal) durante mais de 200 ms. Desta forma, opta-se por módulos do tipo Booksize Compact.

Segundo [13], o cálculo do valor da constante de binário de cada motor - kT é função da própria corrente - I, devido a efeitos de saturação:

$$kT(I) = \frac{M_0}{I_0} + \frac{I - I_0}{Imax - I_0} \times (\frac{M_{max}}{I_{max}} - \frac{M_0}{I_0})$$

Devido a isto, o cálculo desta constante para determinar o valor da corrente em cada instante tornar-se-ia morosa e fora do âmbito prático. Como tal, os valores de corrente necessários para alimentar os servomotores apresentados na tabela 3.2, foram retirados do SIZER da Siemens e não foram calculados individualmente.

Tabela 3.2: Correntes efetiva e máxima em cada servomotor e o respetivo módulo de motor escolhido da marca Siemens.

Servomotor	Corrente efetiva (A)	Corrente Máxima (A)	Módulo escolhido
A	0,86	3,30	6SL3420-2TE11-7AA1
B	4,26	7,94	6SL3420-1TE15-0AA1
C	0,40	2,20	6SL3420-2TE11-7AA1
D	1,15	3,43	6SL3420-2TE11-7AA1

Os modelos dos módulos de motor escolhidos estão representados na tabela 3.3 que, ao comparar com a tabela 3.2, permite entender que os módulos são capazes

[10]Por sobrecarga, enteda-se funcionamento a valores de corrente acima da corrente nominal por períodos intermitentes.

de fornecer corrente suficiente para esta aplicação. Adicionalmente, os módulos 6SL3420-2TE11-7AA1 são duplos, ou seja, permitem alimentar dois motores de cada vez, resultando num total de 3 módulos para 5 servomotores.

Tabela 3.3: Características dos Drives escolhidos da marca Siemens.

Módulo	Corrente nominal (A)	Corrente máxima (A)
6SL3420-1TE15-0AA1	5,0	15,00
6SL3420-2TE11-7AA1	1,7	5,10

Para dimensionamento do módulo de linha, é necessário analisar as necessidades de potência da aplicação, tendo em conta o possível acionamento simultâneo de vários servomotores. A Siemens disponibiliza vários tipos de módulos de linha, e neste caso escolhe-se utilizar um do tipo Smart Line Module porque permite devolver parte da energia da frenagem à rede, uma prática padrão da empresa. A utilização de uma fonte regenerativa terá tanto maior benefício, quanto maior for a potência instalada.

O SIZER faz as contas da potência máxima e potência nominal necessárias a serem fornecidas pelo módulo de linha, tendo em conta o rendimento dos servomotores, dos respetivos módulos e etapas tais como frenagens com o efeito regenerativo. Caso todos os servomotores fossem acionados ao mesmo tempo, resultaria numa potência de pico de 4,98 KW, o que definitivamente não acontece. No entanto, visto que a potência nominal do módulo de linha de menor capacidade disponível (6SL3130-6AE15-0AB1) é 5 KW, a escolha não requer cálculos adicionais, sendo este o selecionado.

A unidade de controlo Siemens escolhida é a CU 320-2, porque se trata apenas de tarefas de posicionamento e controlo de velocidade. Para tarefas de controlo mais exigentes e de maior *performance*, seriam necessárias unidades de controlo da familia SIMOTION D [16].

Outros componentes a selecionar são o filtro de linha (6SL3000-0HE15-0AA0) e o reator de linha (6SL3000-0CE15-0AA0), que são escolhidos em função da potência escolhida para o módulo de linha, pelo que não são necessários cálculos adicionais.

Durante o ciclo, e de acordo com a documentação do SIZER, quando se tem um módulo de linha com capacidades regenerativas, a energia regenerada pelos servomotores quando funcionam como geradores é devolvida à rede, e a resistência de frenagem só é utilizada em caso de emergência ou falha de energia. No entanto, a utilização deste componente é na mesma necessária. A resistência máxima permitida e a potência média dissipada já foram calculadas na parte introdutória desta secção: $\Omega_{res-frenagem} <= 276\ \Omega$ e $P_{diss-media} >= 147W$, respetivamente.

A resistência escolhida é então a 6SN1113-1AA00-0DA0, que permite uma potência média de frenagem em regime continuo de 300 W e apresenta uma resistência elétrica de 17 Ω. O módulo de frenagem é o 6SL3100-1AE31-0AB0, que é selecionado dependendo da resistência selecionada. Verifica-se que resistência de menor capacidade disponível excede bastante as necessidades desta aplicação.

Componentes do circuito elétrico de potência - Rockwell Automation

A versão modular de componentes do circuito elétrico de potência da RA é a série Kinetix 5700. Os valores da corrente nos motores que foram retirados do *software* Motion Analyzer e os respetivos módulos escolhidos estão apresentados na tabela 3.4.

Tabela 3.4: Correntes efetiva e máxima em cada servomotor e o respetivo módulo de motor escolhido da marca RA.

Servomotor	Corrente efetiva (A)	Corrente Máxima (A)	Módulo escolhido
A	0,65	2,54	2198-D006-ERS3
B	4,81	10,31	2198-D012-ERS3
C	0,16	0,9	2198-D006-ERS3
D	1,87	4,62	2198-D006-ERS3

As especificações dos módulos de motor escolhidos estão representadas na tabela 3.5, que permite entender que os módulos são capazes de fornecer corrente suficiente para esta aplicação. Adicionalmente, os módulos 2198-D006-ERS3 são duplos, ou seja, permitem alimentar dois motores resultando novamente num total de 3 módulos para 5 servomotores.

Tabela 3.5: Características dos Drives escolhida para a marca RA.

Módulo	Corrente nominal (A)	Corrente máxima (A)
2198-D006-ERS3	3,5	8,5
2198-D012-ERS3	7,1	16,90

O dimensionamento da fonte de alimentação cai no mesmo caso que para a Siemens: a fonte de menor potência tem uma potência nominal que excede a soma das potências máximas dos servomotores, pelo que se escolhe a de referência 2198-P031[11].

O filtro de linha (2198-DBR20-F) e o reator de linha (1321-3R12-B) são escolhidos em função da fonte de alimentação escolhida, pelo que não são necessários cálculos adicionais.

Por último, a resistência de frenagem é escolhida com base nos cálculos já feitos na subsecção anterior. Tal como na Siemens, o componente com menor capacidade disponível é suficiente e corresponde ao código de catálogo 2198-R004. A sua resistência elétrica é 33 Ω e a potência em regime permanente 400 W (compatível com os valores limite calculados de $\Omega_{res-frenagem} <= 276\ \Omega$ e $P_{diss-media} >= 147W$, respetivamente).

Dá-se assim por concluída a seleção de componentes do circuito de potência para ambas as marcas.

[11]No caso da RA não se trata de uma fonte regenerativa, porque estas apenas estão disponíveis para potências mais altas.

3.2 Componentes do circuito de comando

Nesta secção serão selecionados os componentes que dizem respeito ao circuito de comando da máquina. A partir daqui, toma-se a decisão de apenas fazer a seleção como se existisse um cliente que tivesse optado por componentes da RA. Esta decisão apenas terá impacto na escolha do controlador da máquina e do *software* de SCADA.

Os componentes a selecionar, especificados pela empresa, são:

- PLC de controlo da máquina;
- Codificador angular da posição da mesa;
- *Splitter* do sinal do codificador;
- Computador industrial;
- *Software* de SCADA;
- Fototransístores;
- Barreira de medição.

Será ainda apresentado ao longo desta secção um componente que poderia substituir o codificador da mesa e o *splitter*, sugerido pelo autor.

O modelo de câmara utilizado não faz parte dos componentes que devem ser selecionados. A pessoa encarregue de fazer esta seleção deverá ser alguém responsável pela configuração e programação da interface com a câmara. Caso contrário, a falta de sensibilidade do autor neste aspeto poderia levar a escolhas menos acertadas.

À medida que cada um dos seguintes componentes é apresentado, será feita uma descrição do seu funcionamento e apresentadas as decisões tomadas que resultaram na escolha de cada modelo. Serão incluídas referências ao longo do texto com a devida informação técnica dos componentes escolhidos.

3.2.1 PLC

O primeiro passo para a escolha de um controlador para a máquina é o conhecimento das famílias existentes de controladores. As famílias disponíveis de PLCs da RA são:

- Contrologix - Utilizados para grandes aplicações, são os controladores de topo;
- Compactlogix - Para aplicações médias, que é o caso;
- Micrologix - Para pequenas aplicações.

Dentro da família Compactlogix, existem duas subfamílias: a 5370 e a 5380 (com códigos de catálogo 1769 e 5069, respetivamente). As principais vantagens da família 5380 é o facto de permitir atingir velocidades de comunicação até 1 Gb/s (enquanto para a família 5370 é 100 Mbps) e o processador ser cinco a vinte vezes mais rápido, permitindo controlo de mais eixos, mais nós permitidos na rede EtherNet/IP e leitura mais rápida dos I/Os. Estes motivos levaram a que se optasse por um PLC 5380.



Figura 3.6: Controlador da gama CompactLogix 5380, com alguns módulos de I/O encaixados.

Estes PLCs consistem num controlador ao qual podem ser encaixados vários módulos de entradas e/ou saídas (chamados de módulos de I/Os) lateralmente. Adicionalmente, contêm duas portas distintas para interface com o exterior: porta USB e *ethernet*. A primeira serve para programação do controlador e atualização do seu *firmware*; a segunda serve para comunicar com vários dispositivos tais como computadores, módulos remotos de I/O e *drives* de motores.

Para obter requisitos e selecionar um PLC, é necessário fazer algumas considerações sobre a sua aplicação prevista. Os parâmetros a analisar são: o número permitido de nós na rede EtherNet/IP, a memória necessária do controlador para guardar todo o programa desenvolvido e o número de eixos a controlar. Relembra-se que se considerou que as funções de segurança a implementar são suficientemente simples para poder utilizar um relé de segurança, descartando a hipótese de utilizar um PLC com capacidade de realizar tarefas de segurança.

A quantidade de nós permitidos na rede EtherNet/IP não pode ser estimado à partida porque o número necessário de módulos remotos de I/O é uma incógnita. No entanto, assume-se que ao dimensionar corretamente a memória do controlador, o número de nós permitido será suficiente para esta aplicação. Esta questão ficará mais clara para o leitor quando verificar que o controlador selecionado tem um número bastante elevado de nós permitidos.

Quando o programa desenvolvido no Studio 5000 se encontrava numa fase terminal, o autor aferiu que seriam necessários pelo menos 1,5 MB de memória interna. Considerando que devem sobrar 25 % de memória livre para futuras atualizações e mais 25 % para a finalização programa quando a máquina estiver construída, estima-se uma necessidade de 3 MB de memória.

O controlador deve ser capaz de comandar os 5 eixos previstos para esta máquina. Visto que o número de eixos possíveis de controlar está disponível em potências de dois (2, 4, 8, 16), então será necessário um PLC capaz de controlar oito eixos, no mínimo.

O controlador escolhido é aquele que garante todos estes requisitos ao mesmo tempo. Como não é possível personalizar a configuração de um PLC, o controlador selecionado pode vir a ter maior capacidade do que aquela que é necessária em certos aspetos.

Por outras palavras, o único PLC disponível com 3 MB de memória é o modelo 5069-L330ERM. Este controlador permite até 60 nodos na rede EtherNet/IP, controlar 16 eixos e tem capacidade para 31 módulos de I/O. O aspeto físico deste controlador, com vários módulos de I/O encaixados lateralmente está representado na figura 3.6

3.2.2 Codificador para a câmara

As câmaras de visão do tipo Gocator da LMI technologies (que se utilizarão nesta máquina) necessitam de um codificador (ou *encoder*) para poder associar uma posição angular do pneu a uma linha de pontos da sua superfície, permitindo obter uma nuvem de pontos correspondente à superfície do pneu. O sinal do codificador que é enviado para a câmara deve ser de acordo com o protocolo da camada física RS-485 e conter os sinais A+/A-, B+/B-, Z+/Z- [20].

Para esta aplicação será utilizado um codificador incremental de quadratura (necessário para fornecer os sinais referidos). Um codificador de quadratura é constituído por um disco com três pistas que correspondem aos canais A, B e Z e um par *Light Emitting Diode* (LED)/Fototransístor colocado sobre cada uma das pistas. Cada pista consiste num conjunto de furos igualmente espaçados dispostos em série, que ora permitem a passagem, ora bloqueiam a luz emitida pelo LED.

Desta forma, é possível distinguir dois estados de cada fototransístor - *High* e *Low*. As pistas A e B são dois sinais em quadratura (eletricamente desfasados de 90°), e a pista Z diz respeito à referenciação da origem. Ao detetar as mudanças de estado dos fototransístores sobre as pistas A e B, é possível determinar a posição angular e o sentido da rotação.

O RS-485 é um protocolo de comunicação série para transmissão de dados em alta velocidade e/ou longas distâncias. Cada sinal é transmitido num par de fios entrançados para alcançar imunidade ao ruído: um dos fios contém o sinal e o outro contém o mesmo sinal negado para que possa ser feita uma medição diferencial, cancelando o ruído de modo comum[12]. O RS-485 é geralmente implementado em modo *half-duplex* e permite a presença de até 32 conjuntos transmissor/recetor na rede.

O protocolo RS-422 (no qual o RS-485 é baseado) difere do anterior no facto de ser geralmente implementado em modo *full-duplex* e permitir a comunicação entre apenas dois conjuntos transmissor/recetor. Este protocolo está aqui a ser referido porque o fabricante do codificador especifica que a sua comunicação é segundo o protocolo RS-422, funcionando como um transmissor. A câmara funciona como recetor, pelo que apesar de os fabricantes destes equipamentos referirem protocolos de nomes diferentes, a sua implementação é apenas ao nível da camada física, e os dispositivos são compatíveis.

[12]Corresponde ao tipo de ruído que se propaga da mesma maneira em ambas as linhas. Diferencia-se do ruído de modo diferencial.

Para facilitar a compreensão daquilo que foi acabado de descrever, chama-se a atenção do leitor para a figura 3.7. Na subsecção seguinte volta-se a fazer uma referência a esta figura, e serão esclarecidos alguns detalhes relativamente ao dispositivo identificado como *splitter*.

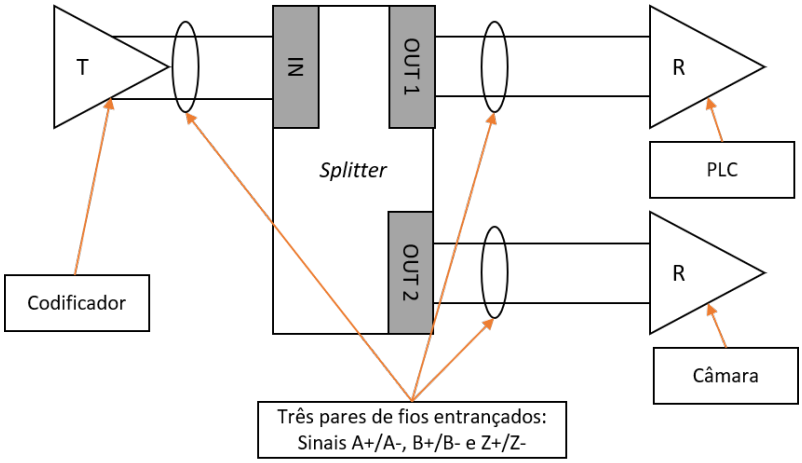


Figura 3.7: Esquema da ligação feita entre o codificador, a câmara e o PLC.

De acordo com [21], a frequência dos pulsos do codificador - f_{pulsos} não deve exceder 1 MHz. Sabendo a velocidade angular da mesa em rotações por segundo - n_{mesa}, a resolução máxima permitida (em termos de pulsos por rotação) - R poderia ser calculada através da equação 3.16:

$$R <= \frac{f_{pulsos}}{n_{mesa}} \tag{3.16}$$

Como ainda não se sabe a verdadeira velocidade angular que a a mesa poderá atingir durante a etapa da câmara, a empresa sugeriu que se escolhesse um codificador com resolução variável e configurável.

Consultando a marca Sick (uma das marcas de eleição da empresa), encontra-se a série DFS60 que apresenta codificadores até 16 bit, com interface elétrico selecionável entre *Transistor-Transistor Logic* (TTL) (RS-422, 5 V ou 24 V DC[13]) e *High Transistor Logic* (HTL) (*Push Pull*). Adicionalmente, pode-se optar por uma funcionalidade de referenciar o zero do codificador através de uma entrada digital, facilitando a sua configuração e fornecendo versatilidade a quem fizer a configuração da câmara.

Estes codificadores estão disponíveis em versão com veio oco ou sólido. Neste caso poder-se-ia optar por um veio oco para uma montagem no eixo do veio da mesa, à saída da caixa redutora do eixo D, através de um acoplamento aparafusado ao veio da mesa. Esta solução tem vantagens no sentido em que não requerer qualquer transmissão, garantindo que a posição angular do veio é a mesma que a posição do codificador.

Alternativamente, poder-se-ia optar por um veio sólido, e montar uma polia no veio da mesa e outra no veio do codificador com diâmetros iguais, ligadas por uma

[13]Os valores 5 V e 24 V dizem respeito à tensão de alimentação, dos quais os 24 V DC serão utilizados.

pequena correia (figura 3.8). Esta solução é mais fácil de conceptualizar e fazer manutenção, pelo que será esta a escolhida.

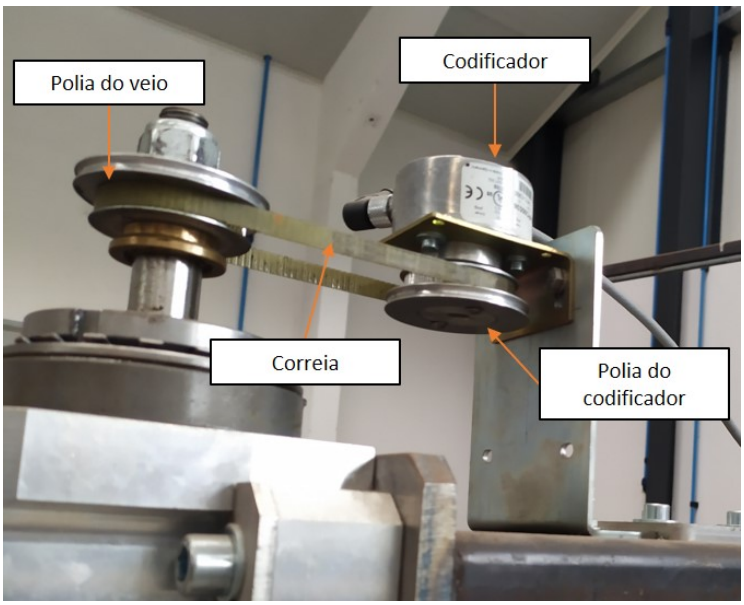


Figura 3.8: Exemplo de uma transmissão do tipo correia - polia entre um veio e um codificador.

As interfaces elétricas disponíveis são o HTL ou TTL. O primeiro significa que a saída não é segundo as especificações elétricas do RS-422, e o valor *"High"* da saída é igual à tensão de alimentação. Este tipo de interface também é conhecido como *push pull* porque o seu circuito elétrico pode "puxar" ou "empurrar" a tensão de saída entre os 0 V e a tensão de alimentação, pelo que não se adequa.

A interface TTL significa que a saída do codificador consiste em pulsos de 0 a 5 V diferenciais, de acordo com o protocolo da camada física RS-422 (eletricamente compatível com o RS-485, como já foi referido).

Em função dos pontos acabados de referir, opta-se pelo codificador de código DFS60A-S4MA65536, cujo catálogo pode ser consultado em [22].

3.2.3 *Splitter* do sinal do codificador

O sinal de posição angular da mesa fornecido pelo codificador deve ser fornecido à câmara, mas é necessário utilizar um *splitter* para que o PLC também possa ter acesso a este sinal. Um *splitter* é um dispositivo eletrónico cuja entrada é o sinal diferencial vindo do codificador e disponibiliza duas ou mais saídas com o mesmo sinal amplificado (figura 3.7). À partida, pode não ser evidente o porquê de se enviar o sinal do codificador para o PLC, visto que já se sabe a posição da mesa que é dada pela posição do servomotor do eixo D.

No entanto, a leitura deste sinal por parte do PLC permite comparação com o valor de posição da mesa dado pelo servomotor para diagnóstico de erros do codificador. Procurou-se um *splitter* com apenas duas saídas, visto que não se prevê mais nenhum componente que necessite de receber esta informação. A utilização de um

splitter com várias saídas seria útil, por exemplo, caso se quisesse ter duas câmaras a funcionar em paralelo com o sinal de apenas um codificador.



Figura 3.9: *Splitter* GV-204 da Motrona [23].

O modelo escolhido tinha como requisitos ser compatível com o sinal TTL/RS-422, e encontrou-se o modelo GV-204 (figura 3.9) da marca Motrona, disponível em [23]. A sua tensão de alimentação é 24 V DC pelo que é compatível com as fontes de alimentação já existentes no quadro.

3.2.4 Módulo de codificador alternativo

No processo de estudo das várias opções de codificadores existentes, encontrou-se uma alternativa a esta solução. Trata-se de um módulo da RA (código de catálogo 2198 - ABQE) que tem capacidade transformar a variável de posição de um eixo do controlador (seja ele virtual ou real) e criar um sinal de saída, como se fosse um codificador com os sinais (A+/A-, B+/B-, Z+/Z-). Este módulo é denominado por *Encoder Output Emulator Module* e está representado na figura 3.10

É ainda possível configurar a resolução do codificador emulado, a sua posição de referência, se a saída é diferencial e se tem um ou dois sinais de saída. Este dispositivo tem capacidade para gerar o sinal de forma rápida suficiente para que o atraso de fase entre a posição do veio físico e o sinal de posição seja pequeno. Esta capacidade dinâmica é conseguida através da programação no PLC de uma tarefa que ocorre sempre que a tarefa de movimento (em inglês, *motion task*) ocorre.

Assim, a sugestão que do autor seria a criação de um eixo virtual com a razão de transmissão 48,95 relativamente ao eixo D. Este eixo representaria a posição da mesa e a sua saída seria programada com a resolução desejada para ligar à porta da câmara. As instruções de como fazer isto estão disponíveis em [24].

Este dispositivo poder-se-ia revelar vantajoso na medida em que dispensaria a utilização do codificador, do *splitter* e das cartas de entradas rápidas do PLC para leitura do sinal vindo do *splitter*. Adicionalmente, seriam eliminados os problemas criados com a montagem do codificador e seria menos um cabo de sinal a ser ciclicamente dobrado numa esteira.



Figura 3.10: Módulo de saída de encoder [24].

Por outro lado, esta solução tem desvantagens no que toca ao preço elevado deste equipamento[14] e ao facto de esta solução ser específica para a RA, ao passo que a solução atual é independente da marca do PLC.

Para não haver dúvidas, repete-se que este módulo não foi utilizado, é apenas uma sugestão do autor.

3.2.5 Computador Industrial

O computador indústrial, ou *Industrial Personal Computer* (IPC), é outro dos componentes que deve estar presente no quadro elétrico. Nele, irá correr o SCADA da RA, um servidor *Open Platform Communications - Unified Architecture* (OPC-UA), e uma aplicação feita por um colaborador da empresa em C, que trata do processamento de sinal vindo da câmara de visão.

Um servidor OPC-UA é uma aplicação que corre no computador, cuja fonte de dados são variáveis do PLC e podem ser escritas por outras aplicações (clientes) que corram no computador [26]. Isto é extremamente útil porque facilita a programação do PLC relativamente à comunicação com outros dispositivos. Nomeadamente, será utilizado para passar as variáveis de comando e *status* da câmara.

A seleção de um IPC foi feita com base num conjunto de requisitos dados pela empresa. Os requisitos são os seguintes:

- 8 GB de memória *Random Access Memory* (RAM);
- *Central Processing Unit* (CPU) Intel Core i5;
- Disco *Solid State Drive* (SSD).

[14]Preço de venda de 600 €, consultado à data da escrita em [25].

A RAM é um tipo de memória volátil utilizada pelo processador para armazenar informações de programas em processamento. Quanto maior quantidade deste tipo de memória, mais programas poderão correr em simultâneo sem comprometer o desempenho. A quantidade de RAM necessária para cada aplicação é normalmente função da quantidade de programas que é previsto estarem a correr no IPC.

O CPU (ou processador) trata-se de um componente indispensável, responsável pela execução de instruções dadas por programas a uma certa frequência, ou *clock rate*. Um CPU pode ter vários núcleos e cada núcleo pode ter várias *threads*. Sendo um processo constituído por várias tarefas, pode-se dizer que o núcleo é responsável por executar o processo, enquanto as *threads* executam as tarefas.



Figura 3.11: Computador industrial AIIS-3400P (PoE) [27].

O IPC escolhido é o AIIS-3400P (PoE) da marca Advantech (figura 3.11), cuja ficha técnica pode ser consultada em [27]. Na lista abaixo encontram-se as características técnicas do modelo escolhido. A seleção do modelo PoE face ao modelo USB 3.0 significa que tem interface para 4 câmaras com Gigabit Ethernet (1 Gb/s), que é o caso da Gocator.

- CPU : Intel core i5 - 6500 3.2 GH 64-bit quad-core, uma *thread* por núcleo;
- RAM : 8 GB SO-DDR4[15];
- Armazenamento : 256 GB SSD (*Solid State Drives*), tamanho 2.5";
- Sistema Operativo : Windows 10 Enterprise IoT 64 bit.

Optou-se por escolher um IPC com três portas de rede (RJ45) para poder colocar o computador em três rede distintas. A primeira rede é a dos dispositivos da máquina, em que o IPC precisa de comunicar com o LASER e o PLC. A segunda é a da fábrica, nomeadamente para comunicação com o servidor SQL. A terceira rede é dedicada à câmara.

A separação da rede da câmara da rede de comunicação do PLC com os seus dispositivos deve-se ao facto de a câmara poder ter de enviar grandes quantidades de

[15]Corresponde ao tipo de memória RAM.

informação. Dessa forma, se a câmra estivesse incluída na mesma rede dos restantes componentes da máquina, poderia sobrecarregar o tráfego provocando atrasos na comunicação. No futuro, poder-se-ia averiguar a possibilidade de evitar esta separação através do envio seletivo de informação da câmara para o IPC.

Dos componentes escolhidos, é ainda necessário fazer considerações sobre o armazenamento e o sistema operativo considerados.

O SSD serve para armazenamento de ficheiros e do sistema operativo. O facto de ser SSD, significa que a tecnologia utilizada para o acesso à informação não é através de um disco físico e agulhas magnéticas, mas sim de dispositivos semicondutores. Desta forma, é possível atingir velocidades de acesso muito maiores do que os clássicos discos rígidos, limitados pela velocidade de rotação do motor.

O sistema operativo Windows Enterprise é a versão do Windows adequada a este setor industrial que traz funcionalidades especiais para ir de encontro às necessidades específicas das empresas [28]. O facto de a arquitetura ser 64 bit em vez de 32 bit, significa que o processador pode aceder a 2^{64} endereços de memória, ou em termos práticos, que pode ter mais do que 4 GB de RAM. A arquitetura é escolhida para corresponder à arquitetura do processador.

3.2.6 *Software* de SCADA

Para usufruir de alta compatibilidade e facilidade de integração com os PLCs da mesma marca, especificou-se que o *software* de SCADA seria o FactoryTalk da RA. Existem duas versões disponíveis: o *Machine Edition* (ME) e o *Site Edition* (SE). Essencialmente, o primeiro não permite a programação em *Visual Basic for Applications* (VBA), que é necessário para comunicação com o servidor SQL. Como tal, utilizou-se o FactoryTalk SE.

A arquitetura do SCADA considerado é constituída por seis componentes:

1. O servidor da HMI - Onde estão guardados todos os ecrãs de vizualização e receitas;
2. O servidor de dados - Onde é feita a comunicação com uma fonte (normalmente um controlador);
3. O servidor de alarmes - Que gere a informação de alarmes sejam eles oriundos de controladores ou de servidores HMI;
4. O cliente HMI - Que deverá correr em cada um dos computadores que têm uma interface com um utilizador;
5. O FactoryTalk Directory - Que permite visualizar a distribuição desta arquitetura;
6. O FactoryTalk View Studio - Ambiente de desenvolvimento de toda a visualização.

A vantagem desta arquitetura é que permite distribuir os vários componentes por diferentes computadores em rede. Por exemplo, o servidor da HMI pode estar

num computador dedicado, e existirem 6 computadores adicionais onde correm as visualizações (clientes) de cada uma de 6 sub estações de uma máquina, cada um com a sua HMI dedicada.

Nesta aplicação, visto que apenas se escolhe o FactoryTalk SE porque tem capacidade de programação em VBA, será utilizada a arquitetura denominada "Estação local". Nesta arquitetura, os quatro primeiros componentes correm no mesmo computador, neste caso, o IPC já referido.



Figura 3.12: Monitor da gama Versaview 5100 [29].

Adicionalmente, deve ser selecionado um monitor de visualização da HMI. Este, deve ter um aspeto semelhante àquilo que se vê na figura 2.5 e deve ser ligado ao IPC. Optou-se pelos ecrãs da gama Versaview 5100 (figura 3.12) da RA, que tem entrada DVI (compatível com o IPC selecionado) e alimentação 24 V DC.

É comum acontecer que o computador industrial (colocado no interior do quadro elétrico) tenha de ficar demasiado afastado do monitor da HMI, tornando a passagem de um cabo demasiado extenso susceptível a interferências e o sinal seja degradado. Para resolver esse problema, existem duas soluções possíveis: a primeira é utilizar um cabo menos suscetível a ruído permitindo aumentar a distância percorrida pelo de cabo [30] e a segunda é utilizar um repetidor que compense a degradação do sinal [31].

3.2.7 Sensores fotoeléctricos

Nesta máquina serão utilizados vários sensores para detetar objetos, sejam eles um pneu ou um componente da máquina. Nesta subsecção serão selecionados os sensores fotoelétricos (também chamados de fotocélulas) que são colocados lateralmente no transportador para detetar a presença de um pneu durante o processo.

Os sensores fotoelétricos são um componente bastante utilizado na indústria dos pneus porque permitem detetar objetos não metálicos, e têm um grande alcance. Estes sensores podem ser divididos em dois componentes principais: um emissor e

um recetor. O emissor emite um feixe nas gamas infravermelho ou visível, que o recetor deve receber e converter num sinal elétrico [32].

A alteração da intensidade do feixe detetado pelo recetor permite identificar a presença de um objeto. Existem várias configurações do conjunto emissor/recetor, e a que se pretende utilizar neste caso é denominada sensor retrorrefletivo, figura 3.13 . Esta configuração consiste na colocação do emissor e recetor de um lado do transportador e de um espelho (que reflete na direção da fonte) do lado oposto.

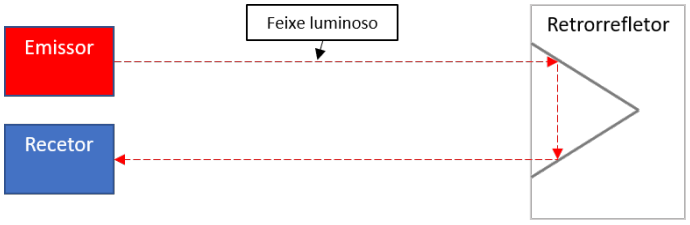


Figura 3.13: Esquema de um sensor fotoelétrico retrorrefletivo.

Quando um objeto opaco passa na dada posição do transportador, o feixe é interrompido e o objeto é detetado. Nesta aplicação, será necessário um sensor com alcance maior ou igual à largura do transportador (1 m). Consultando novamente a marca Sick, optou-se por sensores com as seguintes caracterísitcas:

- Emissão na gama do visível para facilitar a sua calibração;
- Corpo em plástico;
- Conector macho M12 de 4 pinos, visto que não é necessário comprar cabo;
- Saída do tipo PNP, prática comum na automação industrial;
- Método de medição com dupla lente[16].



Figura 3.14: Sensor retrorefletivo da Sick pertencente à família GL-10 [35].

[16]Este método de medição é mais barato que o método *Autocollimation*, e tem como desvantagem a presença de uma pequena zona morta, que não é problemática. As definições de ambos os métodos de medição estão disponíveis em [33] e [34].

O *website* da marca ajuda a fazer a seleção, guiando o utilizador à família Gl-10. A marca disponibiliza uma grande gama de sensores dentro desta família, alguns com capacidade para detetar objetos em situações especiais, como por exemplo envolvidos em papel de alumínio. Como neste caso não é preciso nenhuma dessas funcionalidades, escolheu-se o sensor de modelo Gl-10F4111 (figura 3.14) que está disponível para consulta em [35].

3.2.8 Barreira de medição

O último componente a ser selecionado é a barreira de medição, que será utilizada para medir a largura dos pneus que entram na estação central. Esta leitura é importante por dois motivos: o primeiro é detetar a presença de objetos demasiado altos a entrar na máquina e o segundo é a possível verificação da receita do pneu recebida.

Este equipamento consiste em duas colunas verticais instrumentadas. Na primeira coluna, o emissor, são colocados vários LEDs em série na vertical, que emitem feixes na gama do infravermelho. Na segunda coluna, o recetor, são instalados fototransístores, igualmente dispostos.

Quando um feixe de luz é cortado, o respetivo fototransístor muda o seu estado e assim é possível detetar a presença de um objeto. Ao colocar o recetor e o emissor em lados opostos de um transportador, é possível medir objetos através do sinal dos fototransístores dispostos verticalmente.

Procurou-se escolher uma barreira de medição que tivesse uma saída analógica em corrente 4 - 20 mA, em oposição à escolha de interface de comunicação por EtherNet/IP para não acrescentar mais um nodo à rede e para não obter custos adicionais. Adicionalmente, procurou-se uma barreira com cerca de 450 mm de altura, uma distância entre feixes menor ou igual a 10 mm e distância permitida entre barreiras maior ou igual a 1 m.

As barreiras de medição disponíveis no mercado permitem selecionar modos de funcionamento para várias situações. Por exemplo, um modo de funcionamento para contagem de pequenas peças que interrompam os feixes a altas velocidades ou um modo para medição objetos transparentes. Neste caso, não se requer nenhuma destas capacidades adicionais, pelo que caso houvesse possibilidade de escolha, optar-se-ia por uma barreira que não tivesse esta capacidade.

Ao consultar a Sick, verifica-se que esta marca não disponibiliza nenhuma barreira com saída analógica e sem protocolos adicionais (tais como I/O link, Modbus RTU ou RS-485)[17]. A melhor opção disponível é com comunicação por EtherNet/IP, que também é utilizada por outros componentes, evitando a necessidade de comprar módulos adicionais.

A barreira escolhida é a de código, MLG02N-0445H13201 que tem comunicação por EtherNet/IP, separação entre feixes de 2,5 mm, 455 mm de altura de medição e distância entre barreiras até 2,8 m. Este componente está disponível para consulta em [36], e representado na figura 3.15. Na figura, por baixo da coluna da direita, é

[17] O problema da implementação destes protocolos é ser necessário comprar módulos adicionais para interface com o PLC.

visível um pequeno módulo responsável pela configuração da barreira, interpretação do sinal recebido e comunicação através da rede EtherNet/IP.



Figura 3.15: Barreira de medição da Banner pertencente à família MLG-2 [36].

3.3 Síntese do capítulo

Ao longo deste capítulo, foi feita a seleção de vários componentes de automação da máquina, de acordo com aquilo que foi determinado pela empresa. A primeira parte do capítulo disse respeito à seleção de componentes potência para duas soluções distintas: uma solução Siemens e outra RA.

O dimensionamento de servomotores e respetivas caixas redutoras começou com a descrição das linhas seguidas e métricas calculadas para comparação entre componentes. Depois, foi apresentada a solução encontrada dos conjuntos servomotor/caixa redutora para cada eixo, optando por omitir os cálculos realizados e colocando-os no anexo A. Nesse anexo, é feita uma apresentação em detalhe de cada eixo, do respetivo ciclo de funcionamento e apresentação de todos os cálculos realizados para chegar à solução referida.

Seguiu-se a seleção dos componentes do circuito elétrico de potência necessários para acionar os servomotores de ambas as soluções, através de cálculos realizados com base nos ciclos de funcionamento previstos. Foram selecionados *drives* dos servomotores, e outros componentes que devem ser integrados no circuito para garantir o bom funcionamento e conformidade com as especificações que foram impostas.

A segunda parte do capítulo diz respeito à seleção de alguns componentes do circuito de comando. Começando pelo controlador da máquina, continuou-se para

a seleção do IPC, codificador, *splitter* do sinal do codificador, *software* de SCADA e alguns sensores necessários. Os restantes componentes da máquina que não foram aqui considerados, já haviam sido selecionados pela empresa.

Capítulo 4

Desenvolvimento do programa de controlo da máquina

Após uma apresentação detalhada dos componentes que integram a máquina, é altura de partir para a programação do seu controlador. Antes de entrar diretamente na descrição dos detalhes técnicos e metodologias utilizadas, é necessário introduzir ao leitor o funcionamento dos controladores atuais da RA. Esta primeira secção tocará os aspetos necessários para compreender certas referências feitas sobre particularidades proprietárias, ao longo das restantes secções deste capítulo.

Na segunda secção, será feita uma apresentação técnica e explicação do funcionamento de certos componentes da máquina que terão uma interface lógica com o PLC, nomeadamente o LASER, a câmara, transportadores e servidor SQL. Será descrito o método teórico e prático utilizado para implementar o comando e a sequência lógica destes equipamentos, sem fazer referências ao código-fonte. Esta secção preparará o leitor para a abordagem do comportamento programado da máquina na medida em que ao longo do ciclo existe uma interação necessária entre componentes.

A exposição do comportamento programado começará com a introdução das variáveis mais importantes, criadas para descrever o estado do processo e da máquina. Seguidamente, será apresentada a forma como se comportam e interagem as três estações e os vários estados que podem ser assumidos recorrendo a *grafcets*. Será feita uma análise detalhada das várias rotinas planeadas para atingir os requisitos comportamentais inicialmente previstos.

Depois, será descrita a forma como se estruturou e organizou a programação do PLC e as soluções utilizadas para testar e validar o código desenvolvido.

A última secção consiste na análise detalhada dos ecrãs do SCADA, que foram criados para dar ao operador as ferramentas necessárias para controlar, diagnosticar e comandar esta máquina, e que estão disponíveis no anexo C.

4.1 Introdução aos controladores atuais da Rockwell Automation

Os sistemas de controlo evoluíram ao longo dos anos de forma exponencial: desde a Clepsidra (1400 A.C [37]) ao PLC (1968 D.C), percorreram um longo caminho. A primeira versão daquilo que se entende por sistema de controlo industrial, veio com a invenção do relé que deu asas à lógica cablada. O objetivo de vida de cada relé era

estar integrado num armário elétrico bem recheado de cabos até ao dia de entregar a sua alma ao criador por falha de um dos componentes.

A história conta que em 1960-1970, os colaboradores da General Motors da divisão Hydra-Matic decidiram resolver este problema e desenvolveram um conjunto de critérios para o desenvolvimento de um "controlador de máquina padrão", sujeito a concurso. Quatro empresas responderam à chamada, e a que recebeu o contrato foi a Bedford Associates, com a sua tecnologia chamada MODICON (Modular Digital Control), ainda ressonante hoje em dia. Outra das empresas que concorreram foi a Information Instruments, totalmente adquirida pela Allen-Bradley um ano depois [38].

Em 1976 a Allen-Bradley levou a patente [39] aquilo que chamou de PLC com o modelo Bulletin 1774, cujo sucesso fez com que se tornasse um padrão na indústria. Mais tarde, em 1985 foi comprada pela Rockwell Automation, ficando responsável pela sua divisão de produtos de automação [40].

Hoje, fala-se em *Programmable Automation Controller* (PAC) que diferem dos PLCs na medida em que têm mais do que um processador, são programáveis de acordo com as linguagens referidas na norma IEC 61131-3 (mas não necessariamente todas), permitem controlo de movimento, guardar dados do processo e têm maior capacidade de memória [41; 42]. No entanto, os termos PLC e PAC são normalmente intermutáveis porque não há uma linha clara que separe um tipo de componentes do outro.

Muitas marcas hoje em dia (nomeadamente Siemens e RA) têm a sua estrutura organizada segundo o conceito de arquitetura integrada, que significa a convergência de sistemas de controlo e informação programáveis todos no mesmo *Integrated Development Environment* (IDE) - o Studio 5000 no caso da RA.

Uma das funcionalidades do desenvolvimento em Studio 5000 a referir é a criação de *User-defined Data Types* (UDTs), ou variáveis definidas pelo utilizador. Trata-se uma estrutura de dados, cujos membros podem ser variáveis básicas do PLC, ou outras UDTs, permitindo organizar a informação para condizer com elementos do processo.

Durante a programação de PLCs, é comum necessitar de repetir código para aplicar a situações semelhantes, dentro da mesma aplicação. A título de exemplo, caso se utilizem dois cilindros pneumáticos iguais, faz sentido ter código idêntico que opera sobre I/O diferente. Idealmente, reconhecer-se-ia o cilindro como um objeto, do qual pode haver várias instâncias.

A funcionalidade que se está aqui a introduzir são as AoIs (*Add on Instructions*) que permitem criar um objeto, encapsulando variáveis e funções sob forma de propriedades e métodos e são equivalentes às FBs (*Function Blocks*) da Siemens. A interação com cada objeto é feita através dos seus parâmetros de entrada e saída. A utilização deste tipo de funções é vantajosa em quase todos os aspetos, exceto em um: não permitem edição *online*, e a sua alteração requer um novo *download* para o PLC, o que pode levar alguns minutos.

Um PLC é responsável por executar tarefas programadas. Estas tarefas podem ser divididas em três grupos:

- Periódicas : São executadas periodicamente com uma dada frequência;

- Em eventos : São executadas quando se verifica uma condição;
- Contínuas : São executadas continuamente em *background* com baixa prioridade e sempre que haja possibilidade.

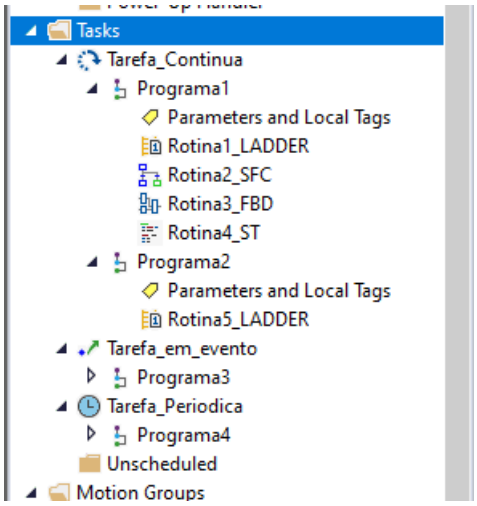


Figura 4.1: Organização hierárquica de tarefas, programas e rotinas num controlador.

Dentro de cada tarefa, podem ser criados vários programas, e dentro de cada programa podem ser criadas várias rotinas em linguagens de programação distintas (figura 4.1) definidas na norma IEC 61131-3[1]. Este tipo de organização permite organizar uma rotina da maneira que o programador achar melhor. Durante o desenvolvimento, vigoram as seguintes regras:

- Cada tarefa tem uma prioridade;
- Programas vistos como prioritários devem estar inseridos em tarefas de maior prioridade;
- Não se deve sobrecarregar computacionalmente o controlador (máximo 32 tarefas);
- A lógica propriamente dita é escrita dentro das rotinas;
- Uma tarefa é divida em programas para associar cada programa a um grupo funcional;
- A cada programa está associado um conjunto de variáveis locais e parâmetros;
- Um programa é dividido em rotinas para permitir associar cada rotina a uma funcionalidade específica.

[1] A IEC 61131-3 contempla a programação de sistemas de automação industriais.

Para trocar informação entre programas, existem duas opções: parâmetros ou escrita em variáveis globais. As variáveis globais estão acessíveis a todos os programas do controlador, pelo que a sua utilização neste sentido deve ser reduzida. Alternativamente, os parâmetros podem ser classificados como:

- Parâmetros de entrada: Passagem de variáveis por valor que são lidas no início da execução do programa;
- Parâmetros de saída: Passagem de variáveis por valor que são disponibilizados no fim da execução do programa;
- Parâmetros de entrada e saída (ou em inglês *Inout parameters*): Passagem de variáveis por referência que mudam de valor por execução do programa;
- Parâmetros públicos: Semelhantes a variáveis globais, mas encapsulados dentro de um programa. A sua passagem é por referência.

Inicialmente, a diferença entre os vários parâmetros pode ser um pouco nebulosa, mas após ter de entender conflitos e erros na programação, torna-se evidente a utilidade de cada um. Nesta aplicação, apenas variáveis que dizem respeito a entradas e saídas constituíram variáveis globais. Todas as outras variáveis foram colocadas dentro de programas, sob a forma de variáveis locais ou parâmetros, dependendo da situação.

Os dispositivos da RA podem utilizar três tipos de topologias de rede [43]:

- DeviceNet: É utilizada para comunicar com, e alimentar dispositivos de campo, tais como sensores, válvulas e variadores de frequência. No entanto, a sua largura de banda é bastante limitada;
- ControlNet: É uma topologia de rede determinística e com capacidade de implementar redundância. Apesar de lenta (máximo de 5 Mbit/s), torna-se ideal para processos que requerem dados em contínuo e sincronizados, tal como controlo de movimento;
- EtherNet/IP: É a topologia mais utilizada, e a única destas três que será implementada na máquina.

Das três, a EtherNet/IP é a mais rápida e com maior facilidade de integração com outro *hardware*. Esta topologia é o resultado da combinação da norma da Ethernet IEEE 802 com o *Common Industrial Protocol* (CIP). Daí o seu nome EtherNet/IP[2], no qual o "IP" significa *Industrial Protocol* e não *Internet Protocol*.

O CIP é o protocolo da camada de aplicação no qual estas três topologias são baseadas. Este protocolo permite aos PLCs comunicarem com centenas de dispositivos de diferentes fabricantes para controlar, sincronizar e diagnosticar. Adicionalmente, também é responsável por permitir que se descarreguem programas de e para o controlador e fazer todo o tipo de edições enquanto o controlador está em processo (também chamadas de edições *online*) [43].

[2]Especificamente com N maiúsculo.

O CIP é orientado a objetos: define que está associada a cada dispositivo uma coleção de objetos que definem as suas características, performance e como ele comunica na rede. Permite ainda implementar sincronização e controlo de movimento através de duas extensões chamadas CIP Sync e CIP Motion, respetivamente [44].

A EtherNet/IP pode recorrer a um de dois protocolos diferentes para a camada de transporte, dependendo da situação:

- UDP (*User Datagram Protocol*):

  É um protocolo utilizado para trocar informação não-crítica e cuja transmissão deve ser rápida e cíclica. Consiste no envio de conjuntos de dados sem verificação de erros ou mecanismos de *handshake*, o que faz com que a entrega não seja garantida. É utilizado, por exemplo, para troca de informação com módulos remotos de I/O [45].

- TCP (*Transmission Control Protocol*):

  Ao contrário do UDP, é usado para trocar informação cuja validade deve ser confirmada. Funciona através do envio de pacotes de informação com os devidos cabeçalhos que, apesar de mais lentos, permitem efetuar verificação e implementar mecanismos de *handshake*. São utilizados em tarefas esporádicas (isto é, não periódicas) como escrita de comandos de referência para dispositivos e no *download* de código para o PLC [45].

Existem ainda outras topologias de rede já utilizadas pela RA mas descontinuadas, que podem ser encontradas *online*.

4.2 Considerações sobre os componentes da máquina

Como já foi referido, a máquina tem três estações: uma estação de entrada, uma central e uma de saída, cujos componentes foram devidamente introduzidos na secção 2.2 - Composição da máquina. Na presente secção pretende-se expor em detalhe todos os aspetos de cada componente funcionalmente dependente da programação e a maneira como esta foi desenvolvida.

Começa-se por fazer uma revisão de todos componentes considerados. Para cada componente, será feita uma descrição da sua constituição física e do seu comportamento.

A estação de entrada é composta por um transportador de rolos com um sensor fotoelétrico em cada extremo. Os pneus devem ser transportados um de cada vez nesta estação e a sua finalidade é servir de interface entre um transportador de pneus já existente na fábrica e a estação central. A estação de saída assemelha-se à estação de entrada em termos de componentes e objetivos.

A estação central é a mais importante da máquina; é dentro desta que se dá o processo de gravação. Em termos de componentes pertinentes para a programação, a lista é constituída por:

- Um portão de entrada materializado através de um cilindro pneumático com sensores nos dois extremos do curso;

- Um sensor fotoelétrico à entrada do transportador, imediatamente antes do portão;
- Um outro sensor fotoelétrico imediatamente após o portão[3];
- O transportador de rolos, com um codificador em um dos rolos para medir o diâmetro externo do pneu[4];
- Uma barreira de medição para medir a largura do pneu (que circula deitado no transportador);
- Uma mesa elevatória acionada verticalmente por um cilindro pneumático com sensores nos dois extremos do curso;
- Quatro eixos A, B, C_{LASER} e C_{camara} com sensores fim de curso e de *homing*, analisados em detalhe no anexo A;
- O eixo D para rotação da mesa, apenas com um sensor de *homing*[5];
- Um portão de saída materializado através de um cilindro pneumático com sensores nos dois extremos do curso;
- Um sensor fotoelétrico à saída do transportador, imediatamente antes do portão;
- Um outro sensor fotoelétrico imediatamente após o portão, no extremo do transportador.

Para facilitar a compreensão do leitor repete-se a figura 2.4 onde são visíveis todos os componentes referidos, exceto a barreira de medição. A programação será feita considerando que a máquina se integrará numa fábrica onde existem:

- Um PLC que controla a linha de transportadores da fábrica e deve comunicar com esta máquina através de um conjunto de entradas e saídas físicas;
- Um servidor SQL, onde devem ser registados os pneus que entram e saem através do preenchimento ou eliminação de linhas em tabelas específicas.

Para resumir toda esta informação, faz-se referência à figura 4.2, onde estão representados todos os componentes com os quais o PLC da máquina interage durante o processo. A sua compreensão é importante para que se compreenda também o ciclo de funcionamento que será descrito na secção 4.3 - Descrição comportamental do funcionamento da máquina.

Ao longo do desenvolvimento de software recorreu-se muitas vezes ao conceito de objeto para identificar componentes que se podem repetir, tais como cilindros pneumáticos, eixos servomotorizados e transportadores. Ao fazer isto, o código torna-se

[3]Utilizado em conjunto com o sensor antes do portão para determinar se há algum objeto a impedir o fecho do portão.

[4]Este codificador não precisou de ser selecionado porque tal não foi especificado pela empresa.

[5]O curso angular deste eixo é de 360°, pelo que não precisa de sensores fim de curso.

quase independente da aplicação, permitindo até copiar de e para outros projetos futuros recorrendo às AoIs, que já foram introduzidas na secção 4.1 - Introdução aos controladores atuais da Rockwell Automation. Como tal, a lógica de controlo dos vários componentes da máquina foi encapsulada dentro de AoIs.

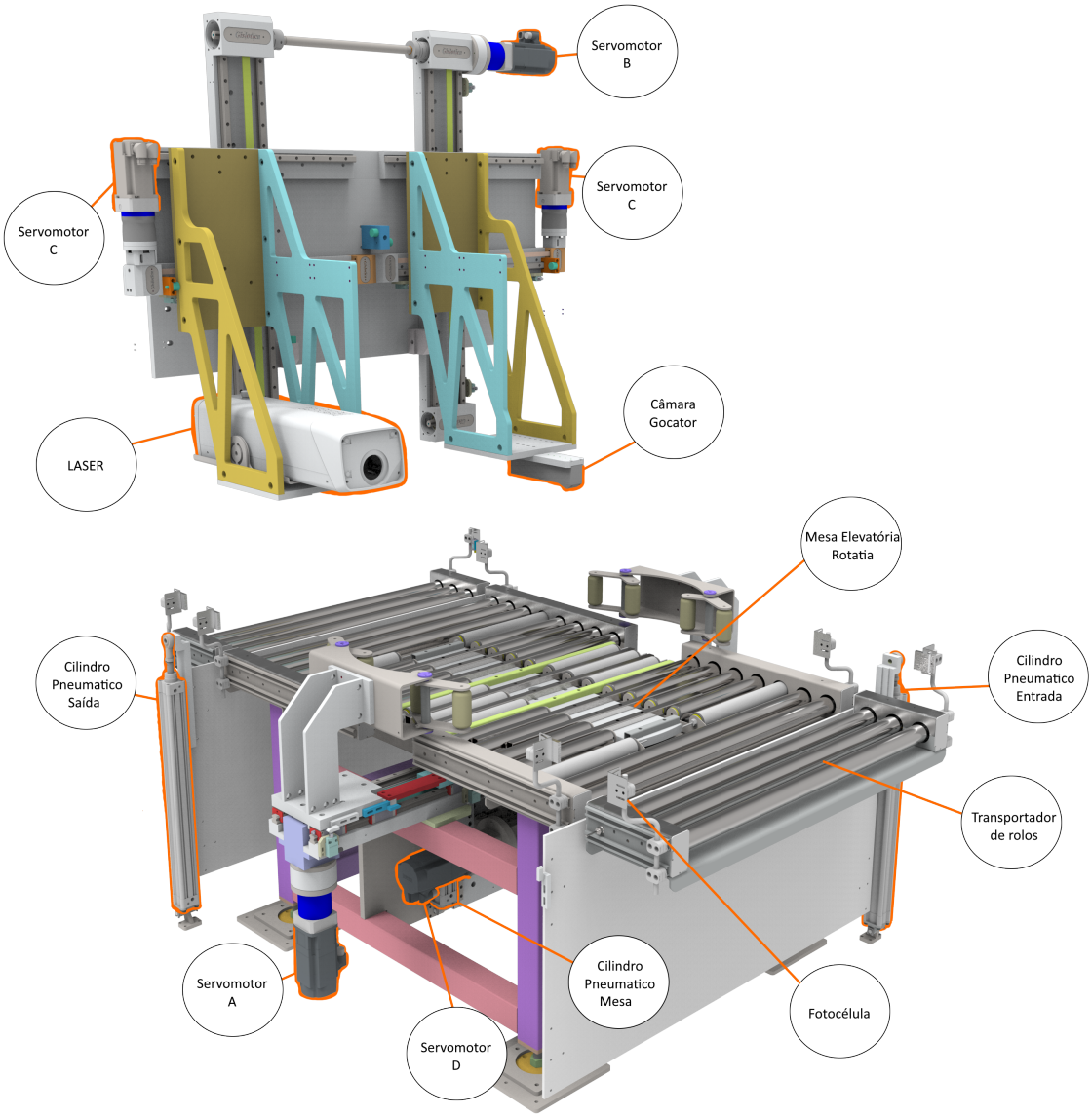


Figura 2.4: Componentes no interior da estação central.Renderização: Gislotica. (Repetida da página 12)

Na aplicação prática da programação de interface do PLC com os componentes aqui apresentados foram incluídas funcionalidades comuns, que devem ser desde já apresentadas. A primeira é o recurso a uma variável de *status*, que consiste numa UDT onde é guardada informação sobre o estado dos componentes, nomeadamente se estão em erro e os seus parâmetros de processo. Em segundo lugar, está a criação uma variável prevista para ser prontamente interpretada no SCADA, para poder controlar o componente ou para consulta do seu estado.

No SCADA FactoryTalk podem ser criados objetos denominados *Global Ojects*,

que constituem mais uma maneira de reduzir a repetição de código e facilitar a integração de vários elementos iguais. Estes *Global Objects* contêm *placeholders* que permitem integrar diretamente uma UDT, em vez de fazer a ligação de várias variáveis dos tipos básicos. Assim, a variável enviada para o SCADA é programada dentro da AoI de cada componente para que ao criar uma nova instância de uma AoI, esta variável seja também automaticamente criada.

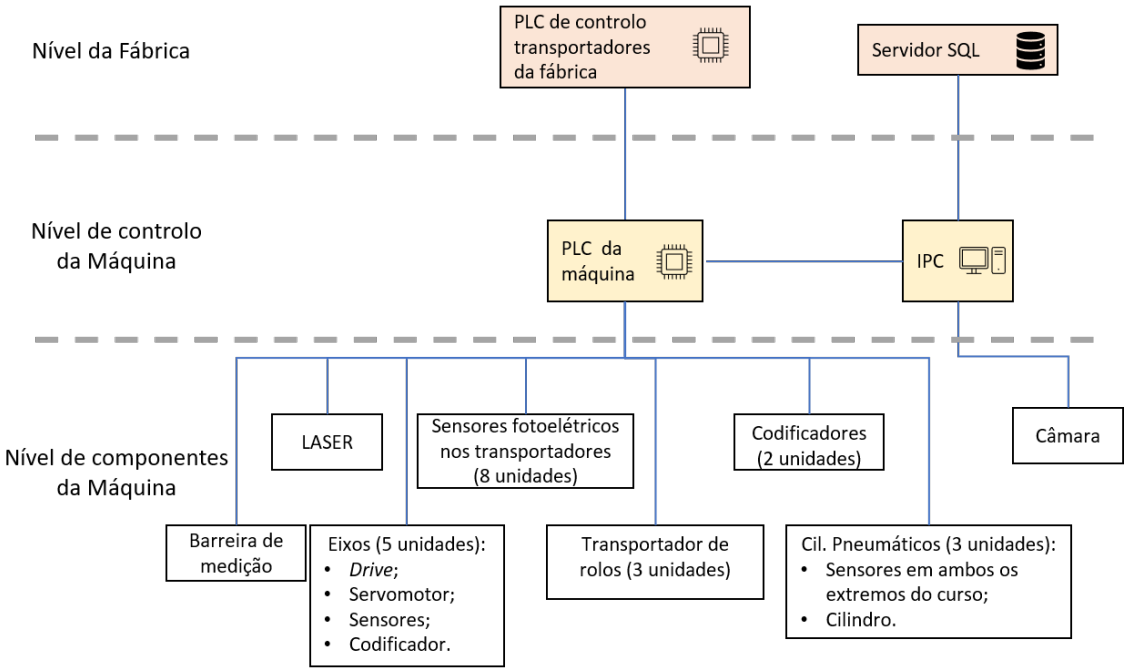


Figura 4.2: Dispositivos que interagem com o PLC da máquina durante o processo.

Existem componentes que têm um determinado comportamento durante o ciclo automático, que não deve ser continuado quando este é interrompido. Nesses casos, utiliza-se o estado da respetiva estação em que se encontra um dado componente como parâmetro de entrada para fazer com que obtenha um comportamento seguro quando se dá uma interrupção. Por exemplo, um servomotor que se encontrava em movimento durante o ciclo é comandado a parar aquando da sua suspensão, sem ter que reprogramar à parte para cada servomotor.

Também foi implementado um parâmetro denominado "*interlock*", que consiste numa variável cujo valor está ciclicamente a ser avaliado pelo PLC em operações lógicas geralmente pouco complexas para que seja de fácil compreensão. Esta variável, presente em quase todos os componentes aqui analisados, serve para inibir o seu comportamento de forma seletiva.

No início da programação, a empresa partilhou um conjunto generoso de AoIs com o autor, que serão devidamente referenciadas ao longo desta secção. Deve-se notar que o objetivo desta secção não é apresentar o código feito, tampouco descrevê-lo exaustivamente. O objetivo é expor a maneira como foi programado o controlo de cada um dos componentes, justificando devidamente.

Após esta introdução, parte-se para a abordagem dos seguintes tópicos:

- Os elementos de comando previstos na HMI;

- A interface com o LASER;
- Comunicação por *sockets* TCP/IP;
- A interface com a câmara;
- A interface com o módulo de controlo dos transportadores;
- A interface com o PLC de controlo dos transportadores da linha;
- A interface com o servidor SQL;
- A interface com cilindros pneumáticos;
- A interface com eixos servomotorizados.

4.2.1 Elementos de comando da HMI

Uma prática comum da empresa é a colocação do painel da HMI num braço adjacente à máquina, permitindo o operador controlá-la e ao mesmo tempo visualizar o processo. Para além do monitor em que está a decorrer a visualização do SCADA, está disponível um conjunto de elementos (tais como botoneiras e seletores) de comando.

Estes elementos são uma excelente maneira de permitir ao operador interagir com a máquina, sem ter de recorrer ao monitor, fazendo parte da tarefa do projetista escolher quais devem ser colocados e a sua função. Muitos deles estão dotados de um *led* no seu interior, cujo estado(*On* ou *Off*) e frequência podem ser codificados para indicar um estado. Esta codificação é normalmente feita para todo um parque de máquinas, fazendo com que a codificação adotada para todas as máquinas seja semelhante, para que não haja equívocos[6].

Na tabela 4.1 são apresentados os elementos escolhidos, o seu nome, a sua cor e a respetiva função. O nome de cada elemento será colocado numa etiqueta colada no painel, e em conjunto com a sua cor, devem permitir subentender a respetiva função. Na secção 4.3 - Descrição comportamental do funcionamento da máquina serão feitas referências aos elementos desta tabela.

Tabela 4.1: Elementos físicos de controlo no painel da HMI.

Elemento	Cor	Função
Botoneira *Start*	Verde	Iniciar o ciclo automático
Botoneira *Stop*	Vermelho	Suspender o ciclo automático
Botoneira *Acknowledge*	Preto	Limpar as variáveis de erro
Botoneira *Move*	Amarelo	Comando sensitivo
Seletor Manual/Automático	Preto	Seleção do modo de funcionamento
Botoneira de emergência	Vermelho	Paragem de emergência

[6]Nesta dissertação, não foi abordada essa codificação, constituindo assim um trabalho necessário no futuro.

4.2.2 Interface com o LASER

Nesta secção será apresentada a maneira como funciona o LASER da marca Koenig & Bauer que será utilizado na máquina (figura 4.3), traçando os aspetos que têm de ser abordados para poder programar devidamente a sua interface com o PLC.



Figura 4.3: LASER da Koenig & Bauer que será utilizado na máquina. Modelo: SPA - C 10.

O LASER comunica com o exterior através de dois meios, cada um com um objetivo diferente:

- Através da rede: Para receber comandos de funcionamento vindos do PLC e para lhe serem configurados os ficheiros a imprimir através do IPC;
- Através de entradas/saídas digitais: Para receber/enviar certos comandos a descrever abaixo.

O princípio de funcionamento deste componente foi consultado nos respetivos manuais de utilização [7] e de comunicações TCP/IP [46], que foram proporcionados pela empresa.

O LASER é um equipamento individual que contém um sistema operativo próprio denominado Scanlinux. Consiste num dispositivo que consegue gravar, a pedido, determinados ficheiros que tenham sido carregados na sua memória interna. Na sua parte traseira dispõe de um conjunto de ligações necessárias ao seu funcionamento:

- Ficha para alimentação monofásica 230 V AC;
- Entrada digital para comando de início;
- Entrada digital para *reset* de alarmes;
- Saída digital para informar que o LASER está pronto a gravar[7];
- Saída digital para informar que a gravação foi concluída;
- Saída digital para informar que está um alarme ativo;
- Entrada digital de *Interlock*: inibe a saída do amplificador do LASER;

[7]Para o LASER estar pronto, é necessário reunir um conjunto de condições de temperatura e estado do dispositivo.

- Entrada digital do Obturador: Fecha o obturador, impedindo a emissão de feixes luminosos para o exterior;
- Porta Ethernet: Para comunicar com a rede de dispositivos do PLC.

A configuração de ficheiros a gravar deve ser feita num computador na rede, através de um *software* proprietário chamado Marca. Cada ficheiro pode ter definidos *placeholders* de texto e numéricos, cuja correspondência é feita conforme uma tabela também programada. A colocação de imagens na memória e criação de tabelas é feita pelo departamento responsável na empresa do cliente.

A comunicação do PLC com o LASER é feita através de sockets TCP/IP. Este tipo de comunicação é definida por um cliente (PLC) e um servidor (LASER), cada um com o seu IP e porta (ou porto) que trocam informações através do envio de pacotes. A maneira como este tipo de comunicação é implementada no PLC será apresentada na subsecção seguinte.

As mensagens que o LASER interpreta nunca são em forma de texto escrito, mas em forma de bytes. Um conjunto de bytes com um significado específico é chamado de trama. O programador do PLC é responsável por criar as várias mensagens através da concatenação de tramas.

A título de exemplo, imagine-se que se quer sondar o tempo marcado no relógio interno do LASER. Neste caso, deve-se enviar uma mensagem com a sequência de bytes (representação hexadecimal) apresentada na tabela 4.2. A respetiva resposta dada pelo LASER está apresentada na tabela 4.3.

Tabela 4.2: Mensagem enviada do PLC para o LASER a pedir as horas do seu relógio interno.

Ordem	Dimensão	Valor	Significado
1	Byte	02	*Start of text*
2	Byte	02	Código específico de comando
3	Byte	38	Código específico de comando
4	Byte	00	Código específico de comando
5	Byte	03	*End of text*

Tal como no exemplo dado, outros comandos podem ser enviados ao LASER. Os valores 0xXXXX-XXXX representam 8 dígitos hexadecimais, e correspondem a um número com 32 bit. Adicionalmente, o LASER envia as *double words* com os bytes trocados, ou seja, a ordem dos bytes está de acordo com a representação *Little Endian*[8].

Para realizar uma gravação, é necessário o LASER estar no modo "gravação". Para tal acontecer, deve-se enviar-lhe uma mensagem com o respetivo comando, no qual se identifica o nome do ficheiro a imprimir e que deve já estar previamente guardado na sua memória. Neste caso, as mensagens são enviadas byte a byte, pelo que os carateres do nome do ficheiro são codificados segundo o seu código ASCII, fazendo corresponder cada carater a um byte em formato hexadecimal.

[8]Esta informação é importante durante a interpretação das tramas por parte do PLC e significa que os bytes são colocados do menos significativo para o mais significativo. Note-se que um byte contém dois dígitos hexadecimais.

Tabela 4.3: Mensagem enviada do LASER para o PLC a responder ao pedido das horas do seu relógio interno, neste exemplo é dia 6/11/1998 e são 15:55:59.

Ordem	Dimensão	Valor	Significado
1	Byte	02	*Start of text*
2	Byte	1E	Código específico de comando
3	Byte	38	Código específico de comando
4	Byte	00	Código específico de comando
5	Double Word	0x06000000	Dia do mês (1 a 31)
6	Double Word	0x0B000000	Mês (1-12)
7	Double Word	0x12000000	Número de anos passados desde 1980
8	Double Word	0xXXXX-XXXX	Dia da semana (1 a 7, não utilizado)
9	Double Word	0x0F000000	Hora (0-23)
10	Double Word	0x37000000	Minuto (0-59)
11	Double Word	0x3B000000	Segundo (0-59)
12	Byte	03	*End of text*

Após o LASER entrar no modo de gravação, é necessário enviar-lhe um comando através da sua entrada digital para comando de início. No entanto, isto só terá efeito desde que o LASER já esteja pronto para gravação. Finda a gravação, o LASER emite um sinal através da sua saída digital que informa que a gravação foi concluída.

Um outro comando bastante útil é o pedido para obter o *status* do LASER. A partir dele, é possível saber se o dispositivo de facto entrou no modo de gravação, se tem algum alarme ativo e que ficheiro é que está a gravar. Desta forma, é possível saber, a pedido, o estado deste equipamento.

A entrada digital do *interlock* serve para inibir a saída do drive em etapas de não funcionamento. Em caso de emergência, deve-se fechar o seu obturador para impedir qualquer emissão de radiação.

Durante o ciclo de gravação do LASER, será preciso calcular as posições tomadas pelos eixos B (vertical), C (horizontal - LASER) e D (mesa rotativa) para que o LASER fique posicionado para efetuar a gravação. Pede-se ao leitor que atente ao anexo B, onde está esquematizado o *layout* dos componentes da máquina e dimensões necessárias para efetuar os cálculos. Estas dimensões serão obtidas por medição direta, após a máquina estar construída.

O objetivo é colocar o LASER por cima da posição onde é suposto fazer a gravação, representada por uma estrela azul no esquema do anexo referido. Para conseguir isso, é preciso posicionar ambos a mesa e o LASER. A posição de trabalho do eixo B (vertical) durante a rotina do LASER - z_L é dada por:

$$z_L = T_L + W_P + H_M - dZ_L \tag{4.1}$$

A variável T_L é obtida do manual de utilização do LASER, e representa a distância entre o ponto a marcar e a lente do LASER. Esta distância varia de lente para lente: a título de exemplo, para uma lente com área de marcação de 100 x 100, deve-se usar $T_L = 160$ mm. O significado da restantes variáveis pode ser consultado no próprio anexo.

A posição de trabalho do eixo D (mesa rotativa) - θ_{M} é calculada através da equação 4.2.

$$\theta_{\mathrm{M}} = 360 + \arccos\left(\frac{dX_{\mathrm{L}}}{\phi_{\mathrm{marc}}/2}\right) - \alpha_{\text{ref-marca}} - \alpha_{\mathrm{ref}} \tag{4.2}$$

Note-se que a posição da mesa é medida no sentido contrário aos ponteiros do relógio. A variável $\alpha_{\text{ref-marca}}$ é um dos parâmetros de entrada vindos do servidor SQL, enquanto a α_{ref} é obtida durante a rotina da câmara Gocator. A posição de trabalho do eixo C (horizontal LASER) durante a rotina do LASER - y_L é dada pela equação 4.3. Para obter essa equação, faz-se as contas partindo do princípio de que o LASER será posicionado por cima do ponto resultante da interseção entre o plano percorrido pelo LASER e uma circunferência de diâmetro ϕ_{marc}[9].

$$y_L = -dY_L + \sqrt{(\phi_{marc}/2)^2 - {dX_L}^2} \tag{4.3}$$

Ficam assim definidas as posições a tomar pelos respetivos eixos durante a gravação a LASER do pneu. Nos próximos parágrafos será abordada a implementação prática do funcionamento do LASER acima descrito.

Visto que se poderiam ter vários LASERs, decidiu-se encapsular o código numa AoI. Deve-se apontar desde já que a metodologia que será agora descrita foi inspirada num exemplo semelhante dado pela empresa para comunicação de um LASER com outra máquina.

Ciclicamente (por exemplo, a cada 5 s), o PLC faz um pedido ao LASER para obter o seu estado. A resposta é dada imediatamente pelo LASER, sendo devidamente interpretada byte a byte pelo PLC, que escreve numa variável interna de *status*.

O *status* deste equipamento consiste numa descrição de todo o seu estado atual, que inclui alarmes ativos, ficheiro a ser gravado, modo de funcionamento e estado da impressão. A posse desta informação por parte do PLC permite saber em tempo útil a reação do LASER aos seus comandos.

Durante o ciclo, é necessário enviar vários comandos, nomeadamente o comando de entrada/saída no modo de gravação. O comando é enviado numa dada etapa do ciclo da máquina através de um booleano que constitui um parâmetro de entrada da AoI. Por sua vez, a AoI faz com que seja enviado o devido conjunto de bytes para o LASER. Ao receber, o LASER responde imediatamente com uma confirmação da ordem, que é prontamente recebida e interpretada nesta AoI.

Para confirmar se o LASER realmente entrou/saiu do modo de impressão, optou-se por esperar pela receção da atualização seguinte do seu *status*. Utilizando o mesmo método, poder-se-iam enviar vários comandos diferentes para o LASER, desde que fossem devidamente configurados segundo o seu manual [46].

4.2.3 Comunicação através de *sockets* TCP/IP

A comunicação por *sockets* é bastante utilizada no contexto industrial porque permite comunicação entre dois dispositivos de diferentes marcas. Nesta aplicação foi

[9]Tecnicamente, esta interseção resulta em dois pontos, mas o plano de movimento do LASER é espacialmente limitado ao curso dos eixos, fazendo com que apenas um dos pontos seja alcançável.

implementada para fazer comunicar o PLC com o LASER, mas também serviria para outras aplicações, tais como:

- Comunicação do PLC com um robô;
- Comunicação do PLC com uma aplicação no computador;
- Comunicação do PLC da RA com um PLC da Siemens (dado que os seus outros protocolos de comunicação são incompatíveis).

O nome *socket* é uma analogia de um encaixe da ficha de um cabo que transmite informação: se o cabo for encaixado em *sockets* de dispositivos nos dois extremos, então eles podem comunicar. Assim, o *socket* é a implementação em *software* do encaixe do cabo, que deve seguir um certo procedimento para que a comunicação seja estabelecida com sucesso por ambas as partes. Na realidade, não precisa propriamente de haver um cabo físico a ligar ambos os dispositivos, basta que estejam em rede um com o outro.

Um *socket* é um objeto disponível no módulo EtherNet/IP que permite criar uma ligação com outro do mesmo tipo presente noutro dispositivo [47]. A comunicação aqui implementada funciona com *Sockets* TCP, segundo o modelo cliente/servidor na medida em que o PLC faz um pedido para estabelecer a comunicação e o LASER fica à espera e apenas tem de aceitar/rejeitar os pedidos. Também existem *sockets* UDP que não são aceites pelo LASER, visto que o envio de informação é feito de maneira diferente e não serão aqui abordados.

Para um *socket* ser funcional, é preciso seguir dois passos:

1. Criar um objeto *socket* no módulo do controlador tanto no cliente como no servidor;
2. O cliente deve abrir a comunicação com o *socket* do servidor (que pode ou não ser aceite);

A comunicação por *sockets* TCP/IP é entre dois dispositivos, sendo cada um identificado por um endereço IP e uma porta (ou porto). Concluídos os dois passos referidos, cada um dos dispositivos pode proceder ao envio/leitura de informação.

O próprio Studio 5000 disponibiliza um conjunto de instruções (MSG) que já têm disponível todas as operações que o programador possa querer fazer. Adicionalmente, existe um documento da própria RA [47] que contém as instruções de utilização destas instruções (MSG) passo-a-passo.

Na prática, programou-se isto sob a forma de uma AoI que permite criar *sockets*, abrir a ligação, eliminá-los e enviar/receber informação através da escrita de variáveis booleanas que constituem os seus parâmetro de entradas. Quando um *socket* perde a conexão, ou entra em erro, isto é devidamente anunciado à rotina de controlo da estação central, e o ciclo de processo automático é suspenso. Quando o operador carregar no botão *acknowledge*, o *socket* expirado é eliminado, seguindo-se a criação e abertura de um novo *socket*.

Deve ser colocada uma destas AoIs por cada dispositivo com o qual o PLC comunique por *Sockets* TCP/IP.

Não foi a melhor opção ter utilizado uma AoI para este efeito porque as instruções do tipo MSG requerem a criação de uma variável global e devem ser configuradas uma a uma. Ao ter as instruções dentro de uma AoI, tornou-se impossível configurá-las em modo *online* e uma nova alteração e teste implicou um novo *download* de todo o programa para o PLC, tornando o processo lento. Em contrapartida, a utilização de uma AoI para encapsular este código condensou-o e tornou-o mais fácil de replicar.

4.2.4 Interface com a câmara

Esta máquina contém uma câmara de visão do tipo Gocator da marca LMI Technologies. Esta câmara utiliza triangulação LASER para fazer a leitura da superfície de um pneu e recolher as coordenadas de vários pontos - uma nuvem de pontos.

Para realizar esta tarefa, a câmara é dotada de um emissor e um recetor, figura 4.4. O emissor emite vários feixes LASER, dispostos em linha reta e na direção radial do pneu. A radiação é refletida e captada pelo recetor, permitindo calcular a distância entre cada ponto da superfície do pneu e a câmara, através de relações geométricas [20].

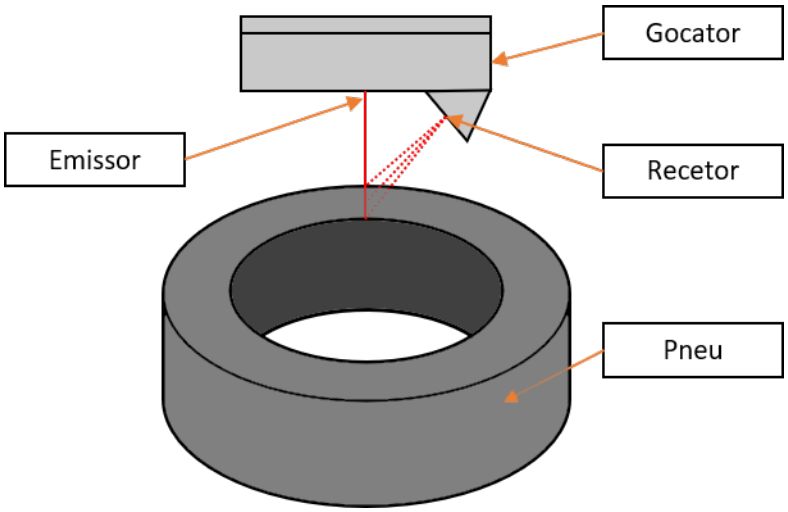


Figura 4.4: Esquema da colocação da câmara relativamente ao pneu.

A distância entre dois pontos consecutivos da superfície do pneu lidos na direção radial é chamada a resolução. Esta métrica depende do número de feixes emitido pela câmara e da distância entre o pneu e a superfície da câmara. Para que a tarefa referida seja realizada ao longo de toda a superfície do pneu, é preciso fazer leituras periódicas à medida que este roda, cuja frequência é chamada de *tick rate*.

Obtendo-se a nuvem de pontos, é possível procurar e identificar elementos específicos da superfície. Por exemplo, pode-se obter informação sobre letras gravadas devido ao seu relevo, ou então a posição de um determinado símbolo indentado na parede do pneu. Neste caso, o objetivo é encontrar um certo símbolo indentado, conhecendo à partida que tipo de figura geométrica é, e quais as suas proporções. Por exemplo, se for um retângulo, poderá ser a figura geométrica número 17, de proporção 2 (a largura é duas vezes maior que a altura).

Na figura 4.5 podem-se identificar os vários componentes que constituem a câmara, atentando agora às suas conexões com o exterior.

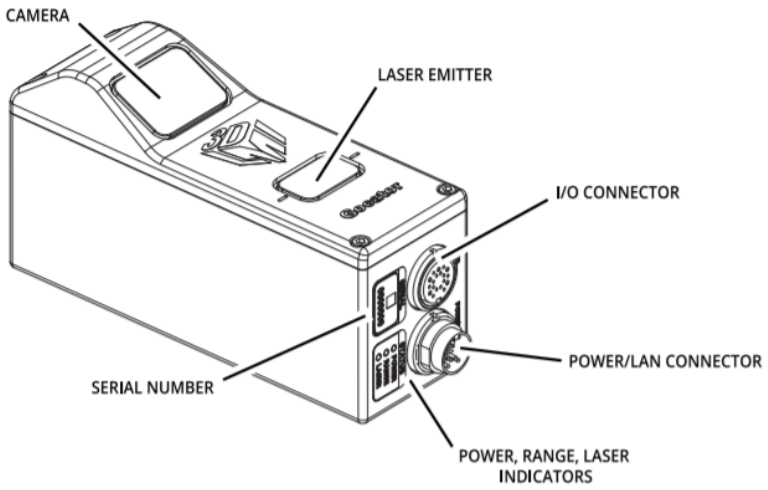


Figura 4.5: Legenda dos componentes externos de uma Gocator [20].

Ligado ao *I/O Connector* está um cabo que pode ter vários sinais com informação do processo, tais como *Interlock*, entradas e saídas digitais, saída analógica e entrada de sinal de um codificador[10] (através de comunicação série RS-485) e alimentação dos sensores. A ficha identificada como *Power / Lan Connector* é utilizada para comunicação Ethernet com velocidade até 1 Gb/s e contém ainda dois fios para alimentação da câmara.

A zona de alcance da câmara está descrita na figura 4.6. Pode-se verificar que a zona ativa de funcionamento constitui um trapézio cujos lados paralelos são denominados *Near Field of View* e *Far Field of View*, e com uma altura de *Measurement Range*. Desta figura, pode-se concluir que a câmara tem de estar posicionada dentro de um certo intervalo de distâncias do pneu.

Esta câmara permite comunicação com PLCs da RA através do protocolo EtherNet/IP, quer por mensagens explícitas (TCP - utilizando a instrução MSG), quer por mensagens implícitas (UDP - diretamente escrevendo nas variáveis do controlador). Independetemente das vantagens/desvantagens entre TCP e UDP, nenhuma destas será utilizada pelo simples facto de que os dados vindos da câmara têm que ser processados, o que não é uma tarefa do PLC.

Visto que estas câmaras são *smart sensors*, também são capazes de processar dados. No entanto, dada a quantidade de informação a ser processada, estar-se-ia a colocar altas exigências no processador deste equipamento. Como tal, opta-se por dedicar as suas capacidades apenas à aquisição de informação, deixando o processamento para outro componente.

O processamento da nuvem de pontos é feito no IPC, com o qual a câmara pode comunicar através do chamado "*Gocator protocol*". Isto é implementado através da utilização de um conjunto de bibliotecas abertas em C, chamadas GoSDK [20]. A integração da câmara na máquina já está a ser realizada pela empresa noutras

[10] O codificador selecionado na subsecção 3.2.2 - Codificador para a câmara do capítulo 3.

máquinas; como tal, ao longo da programação do PLC, a câmara foi considerada um objeto para o qual apenas é necessário enviar comandos básicos (tais como "iniciar processo", "abortar processo" e "pedir estado").

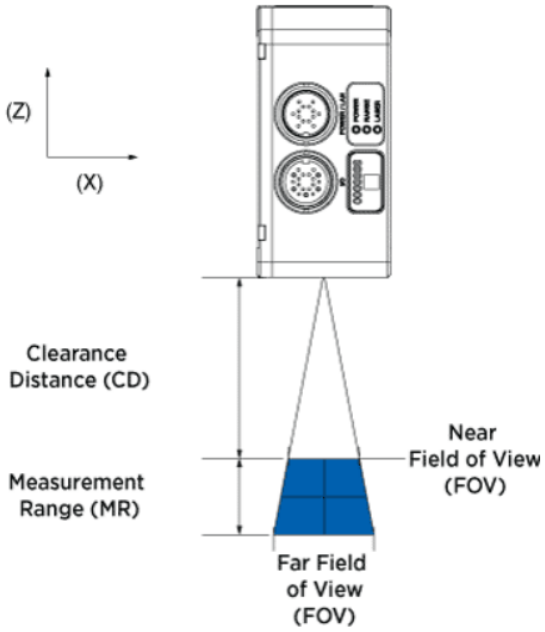


Figura 4.6: Campo de visão de uma Gocator.

A aplicação desenvolvida em C para processamento de dados é cliente de um servidor OPC-UA, cuja fonte de dados é um um conjunto de variáveis do PLC. Alternativamente, a aplicação em C que corre no computador também poderia comunicar com o PLC através de *sockets* TCP/IP, o que aumentaria a complexidade da programação porque se estaria a fazer processamento de bytes, tal como é feito para o LASER. Nesse caso, se se quisesse fazer alterações na comunicação, ter-se-ia que criar uma nova mensagem que é sempre temporalmente dispendioso porque obriga a uma reprogramação correspondente do lado do PLC.

Do ponto de vista do PLC, a câmara é um objeto com parâmetros de entrada e saída. Atente-se à tabela 4.4, onde estão representados e descritos os parâmetros de entrada e saída da câmara, como objeto. Alguns dos parâmetros são apenas propriedades, isto é, são variáveis que servem para passar e guardar informação; outros são comandos para execução de métodos (ou funções) da câmara.

Segue-se um exemplo para explicitar como é que o PLC dá ordens à câmara:

Pretende-se dar uma ordem de início de ciclo de procura da referência do pneu com um determinado tipo de referência e um determinado rácio largura/altura. Para isso, o PLC deve escrever nas variáveis de entrada *Tipo de referência* e *Rácio largura/altura* os valores desejados, seguido de uma transição ascendente da variável *Comando início*. Ao detetar isto, a câmara saberá que deve executar a respetiva rotina.

Durante o ciclo da câmara, será preciso calcular as posições tomadas pelos eixos B (vertical) e C (horizontal - câmara) para que esta fique posicionada por cima do diâmetro da referência que deve encontrar, representada por um losango azul no

Tabela 4.4: Parâmetros de entrada e saída da câmara.

Nome	Tipo de variável	Descrição
Entradas		
Comando início	Booleano	*True*: Ordem de iniciar o ciclo de procura da referência
Comando abortar	Booleano	*True*: Ordem de abortar o ciclo de procura da referência
Tipo de referência	Inteiro de 32 bit	Tipo de referência a procurar no pneu
Rácio largura/altura	Inteiro de 32 bit	Razão entre a largura e altura da referência a procurar
Saídas		
Conectado	Booleano	*True*: Câmara conectada
Pronto	Booleano	*True*: Câmara pronta a realizar o seu ciclo
Em processo	Booleano	*True*: Ciclo em processo.
Sucesso	Booleano	*True*: Ciclo concluído com sucesso
Abortado	Booleano	*True*: Ciclo abortado com sucesso
Código de erro	Inteiro de 32 bit	Código de erro atual na câmara
Ângulo referência-zero	Real	Ângulo entre o zero do codificador e a posição da referência

anexo B, onde está esquematizado o *layout* dos componentes da máquina e dimensões necessárias para efetuar os cálculos. Estas dimensões serão obtidas por medição direta após a máquina estar construída.

A posição de trabalho do eixo B (vertical) durante a rotina da câmara - z_C é dada por:

$$z_C = T_C + W_P + H_M - dZ_C \tag{4.4}$$

A variável que representa a distância entre o ponto a marcar e o emissor da câmara - T_C é obtida do manual de utilização, em função da resolução que se pretende obter. A distância que será efetivamente utilizada é normalmente escolhida e otimizada na fase de testes.

Durante o ciclo da câmara, a mesa é rodada pelo servomotor D enquanto a câmara procura a referência. A posição de trabalho do eixo C (horizontal câmara) durante esta rotina y_C é dada pela equação 4.5. Para obter essa equação, faz-se as contas partindo do princípio de que a câmara será posicionada por cima da circunferência criada pelos pontos onde a referência se pode encontrar, cujo diâmetro - ϕ_{ref} é obtido do servidor SQL.

$$y_C = -dY_C + \sqrt{(\phi_{ref}/2)^2 - {dX_C}^2} \tag{4.5}$$

Fica assim explicada a maneira como a câmara Gocator se integra e comunica com o PLC da máquina, e as posições a tomar pelos respetivos eixos durante a sua rotina de identificação da referência. Neste caso, não houve necessidade de criar uma AoI em específico porque o teste deste componente não foi possível realizar.

4.2.5 Interface com o transportador e Multicontrol

Um dos requisitos impostos foi que se utilizassem rolos da marca Interrol para os transportadores das estações, do modelo EC 310, cujo aspeto exterior está apresentado na figura 2.4 e o manual técnico referenciado em [48]. Estes componentes

aparentam ser rolos normais por fora, mas podem ter um motor DC e caixa redutora embebidos, que tornam o conjunto bastante compacto e de fácil manutenção. Isto é especialmente utilizado em aplicações limitadas em termos de espaço e em que as cargas não sejam demasiado elevadas.

A alimentação e controlo destes rolos motorizados é feita através de um pequeno módulo denominado Multicontrol (figura 4.7). Este módulo permite ligação de até quatro rolos e ligação de até oito entradas digitais, das quais quatro podem ser configuradas para funcionar alternativamente como saídas. Estas entradas/saídas permitem, por exemplo, incluir um sensor para detetar a presença de um pneu numa dada secção do transportador ou um sinalizador luminoso.



Figura 4.7: Módulo Multicontrol da Interrol para controlo de rolos.

Cada módulo pode ter até quatro rolos, mas num transportador, nem todos os rolos são motorizados: os rolos que o são, acionam correias da própria marca denominadas Polyvee, permitindo ter cerca de quatro rolos não motorizados por cada rolo motorizado. Desta forma, um módulo Multicontrol pode ser suficiente para acionar todo um transportador.

O módulo tem ainda uma porta de rede para conectar ao PLC, uma alimentação 24 V DC para a eletrónica e outra alimentação 24 V DC para energizar os rolos, permitindo cortar apenas a última em caso de emergência. Os cabos da alimentação são feitos para que a sua integração na máquina seja o mais rápida possível, sem ser preciso descarnar o cabo e colocar ponteiras. Na parte posterior do módulo tem um conjunto de "dentes" que mordem no cabo e fazem imediatamente a ligação elétrica, tornando-se uma solução bastante prática.

O Multicontrol é configurado através do acesso à sua página web, disponível no *browser* quando o computador se encontra na mesma rede. Para aceder, basta escrever o IP do módulo na barra de endereços e aí pode-se proceder à configuração de vários parâmetros, tais como velocidade máxima dos rolos, protocolo de comunicação (neste caso EtherNet/IP), número de rolos comandados, configuração de entradas/saídas e modo de controlo.

Estão disponíveis vários modos de controlo, incluindo situações em que não precisa sequer de existir um PLC. O modo de controlo escolhido nesta aplicação é denominado "I/O mode", no qual os rolos são sempre comandados pelo PLC a partir de um conjunto de entradas/saídas digitais disponibilizadas nas variáveis globais, sob a forma de duas UDTs (uma para entradas e outra para saídas). Estas duas va-

riáveis contêm a informação um pouco desorganizada, mas que pode ser condensada nos seguintes tópicos:

- Velocidade a comandar a cada rolo;
- Valor lido nos sensores conectados;
- Valor de comando das possíveis saídas digitais;
- Erros do módulo e de cada rolo individual;
- *Feedback* da velocidade comandada a cada rolo.

Por exemplo, para controlar a velocidade do rolo número 1, então deve-se escrever na respetiva variável o valor da velocidade a comandar no intervalo [-100 % a +100 %].

Para programar o controlo do Multicontrol foram criadas duas AoIs distintas: a primeira simboliza o próprio módulo Multicontrol e a segunda simboliza o transportador.

A AoI do módulo tem como função pegar nas variáveis globais criadas automaticamente e discretizá-las devidamente em variáveis fáceis de compreender e comandar. Como parâmetros de entrada desta AoI, estão quatro velocidades individuais para comando individual de cada um dos rolos. Deste modo, até poderiam existir quatro transportadores individuais, com um rolo cada um, sendo cada rolo comandado a uma velocidade diferente.

A AoI que simboliza o transportador tem como entradas comandos dados normalmente a um transportador (avançar/recuar) e como saída a variável de velocidade comandada do transportador. É precisamente esta variável que é irá ligar às entradas individuais de cada um dos rolos da AoI do Multicontrol. Note-se que podia facilmente ser configurado um transportador com acumulação de pneus, através da criação de várias instâncias desta AoI: uma por cada pneu a acumular.

Como se pode deduzir, desta forma é possível ter um transportador a utilizar rolos de vários módulos Multicontrol. Também se pode ter um Multicontrol e comandar rolos de vários transportadores diferentes. Esta solução encontrada permite minimizar o número de módulos multicontrol comprados.

Por exemplo, imagine-se que uma máquina tem dois transportadores T1 e T2, que precisam de dois rolos motorizados cada um. Utilizando as AoIs referidas (duas AoIs de transportador e uma de módulo), apenas é preciso comprar um módulo Multicontrol. Esta solução contrasta com o que o autor pensou à partida, que era desenvolver uma única AoI fazendo corresponder um transportador necessariamente a um módulo Multicontrol.

4.2.6 Interface com o PLC de linha

A linha da fábrica onde a máquina será integrada contém vários transportadores, controlados por um controlador aqui chamado de "PLC de linha". Este controlador é responsável pela coordenação do fluxo de material.

Foi especificado que a comunicação entre este e o PLC da máquina fosse feita através de um pequeno conjunto de entradas/saídas digitais. Alternativamente, a comunicação poderia ser feita por rede: através das chamadas variáveis produzidas/consumidas se o PLC de linha também fosse da RA, ou através de *sockets* TCP/IP se fosse de outra marca (por exemplo Siemens). As variáveis trocadas são apresentadas na tabela 4.5, note-se que as entradas são do ponto de vista do PLC da máquina e correspondem a uma saída do PLC de linha.

Tabela 4.5: Entradas e Saídas digitais trocadas com o PLC de linha e o respetivo significado quando apresentam o valor *True*.

Nome	Significado
Entradas	
Transp. Entr. Stop	Transportador antes da estação de entrada não acionado.
Transp. Saida Stop	Transportador depois da estação de saída não acionado.
Pronto Dar	Transportador antes da estação de entrada "pronto a dar" um pneu.
Pronto Receber	Transportador depois da estação de saída "pronto a receber" um pneu.
Pausa Auto.	Ordem de suspensão do ciclo automático.
Saídas	
Est. Entr. Stop	Transportador da estação de entrada não acionado.
Est. saida Stop	Transportador da estação de saída não acionado.
Entr. Pronto Receber	Transportador da estação de entrada "pronto a receber" um pneu.
Saida Pronto Dar	Transportador da estação de saída "pronto a dar" um pneu.

A partir destas nove variáveis é feita a comunicação de estados e o *handshake* entre a máquina e a fábrica. As entradas/saídas que dizem respeito a "pronto a dar" ou "pronto receber" fazem parte do protocolo de transferência de pneus e informação entre a máquina e a fábrica. A sua utilização será explicada na subsecção 4.2.7 - Interface com o servidor SQL.

As entradas/saídas que dizem respeito a sinais de "*Stop*" significam que o respetivo transportador não está a ser energizado. A título de exemplo, imagine-se que se está a fazer uma transferência do transportador da fábrica para a estação de entrada, estando o pneu metade em cada transportador.

Se o transportador da fábrica tiver que parar, então a entrada *Transp. Entr. Stop* será levada a *True*; ao detetar isto, o PLC da máquina desenergizará o transportador da estação de entrada. Para retomar o ciclo, a variável *Transp. Entr. Stop* retorna ao valor de *False*, permitindo voltar a acionar o transportador da estação de entrada. O mesmo pode acontecer caso a máquina precise de desenergizar o seu transportador a meio de uma transferência, através da saída digital *Est. Entr. Stop* que informará o PLC de linha que deve parar de acionar o transportador antes da estação de entrada.

A entrada *Pausa Auto* serve para fazer um pedido à máquina de suspensão do seu ciclo automático de forma remota, isto é, sem que o operador se desloque até à máquina e faça o pedido através da HMI.

Assim, fica apresentada a comunicação feita entre o PLC de controlo dos transportadores da fábrica e o da máquina, que é especialmente necessária para a compreensão do processo de transferência de um pneu.

4.2.7 Interface com o servidor SQL

Na transferência de um pneu do transportador da fábrica para o transportador da máquina (ou vice-versa), deve ser implementado um protocolo que englobe fluxo de material e informação, garantindo que se possa recuperar de uma situação de interrupção sem perda do fluxo. A transferência de informação é executada através da escrita de duas tabelas específicas de um servidor SQL presente na rede da fábrica. Uma das tabelas é relativa à estação de entrada e outra à de saída e serão denominadas tabela de entrada e saída, respetivamente.

Para garantir que durante o processo de transferência de um pneu não ocorrem erros, é preciso definir um protocolo que preveja a ordem dos acontecimentos. Essa ordem de acontecimentos está representada no diagrama temporal da figura 4.8, que passa a ser descrita.

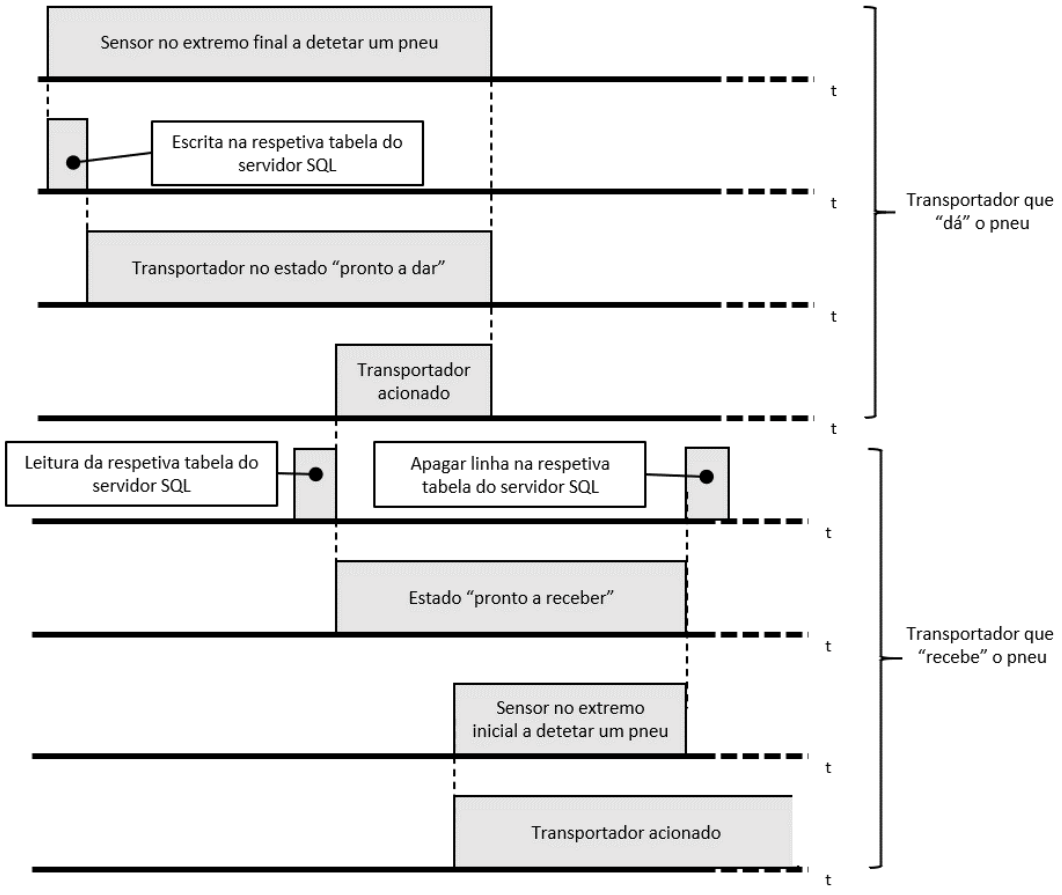


Figura 4.8: Diagrama de estados da transferência de pneu entre um transportador da máquina e um transportador da fábrica.

Para um pneu ser transferido entre dois transportadores, o que fornece o pneu tem de estar no estado denominado "pronto a dar", e o que recebe no estado "pronto a receber". Nesta máquina, o fluxo ocorre sempre da entrada para a saída, pelo que o transportador que "dá" é sempre aquele que está imediatamente antes do transportador que "recebe", no sentido referido.

O transportador que "dá" chega ao estado "pronto a dar" quando é detetada a presença de um pneu no sensor colocado no seu extremo final, e fica à espera de que o transportador que "recebe" entre no estado "pronto a receber", sendo este estado alcançado quando este transportador está pronto para receber um pneu. Quando o que "dá" vê que o seguinte está "pronto a receber", o transportador é acionado até que o sensor no seu extremo final deixe de detetar um objeto. O transportador que recebe apenas sai do seu estado "pronto a receber" ao detetar que já terminou a transferência de um pneu.

Ao transferir um pneu do transportador da fábrica para a máquina, o PLC de linha deve escrever (e confirmar que escreveu) o respetivo ID e receita[11] na tabela de entrada antes de entrar no estado "pronto a dar". Quando a máquina estiver pronta a receber o pneu, antes de entrar no estado "pronto a receber" e se o transportador da fábrica estiver "pronto a dar", então a máquina irá ler a primeira e única linha da tabela de entrada e guardar as informações numa variável. Verificando que o ID do pneu é válido (maior que zero), a máquina entra no estado "pronto a receber", levando ao acionamento do transportador da fábrica.

Tecnicamente, as tabelas de entrada e saída apenas precisariam de ter uma coluna com o ID do pneu a entrar/sair da máquina, visto que se considera que cada pneu na fábrica tem um ID diferente. Depois, utilizar-se-ia o ID para encontrar noutras tabelas informações sobre a respetiva receita, de acordo com a organização da fábrica. No entanto, opta-se por definir que as tabelas de entrada e saída teriam várias colunas já com a receita completa, evitando a necessidade de fazer *queries* a várias tabelas[12].

No instante em que o pneu a ser transferido é detetado pelo sensor no extremo inicial do transportador da estação entrada, o pneu é considerado como dentro da estação de entrada, altura em que o respetivo transportador é acionado. A transferência continua até que o sensor no extremo final do transportador da fábrica deixa de sentir o pneu, deixando de considerar o pneu como dentro do transportador da fábrica, desenergizando-o. O transportador da estação de entrada continua acionado até que o sensor no seu extremo inicial deixe de sentir o pneu, altura em que o *status* do pneu passa a ser considerado como totalmente dentro da estação de entrada, e é enviado um pedido ao servidor SQL para eliminar a respetiva linha da tabela.

O processo de transferência um pneu da estação de saída da máquina para o transportador da fábrica à saída, é feita pelo mesmo processo. Caso o ciclo seja suspenso durante algum dos instantes, não há problema, porque uma suspensão pode sempre ser retomada. No entanto, situações em que é necessário proceder a uma reinicialização da máquina é que são problemáticas porque o pneu pode estar na fronteira entre dois transportadores consecutivos.

Imagine-se que durante a transferência um dos transportadores deixa de funcionar: o outro não pode continuar energizado, porque iria fazer com que o pneu fosse arrastado de maneira imprevisível devido a velocidades diferentes em dois pontos.

[11] A receita é um conjunto de parâmetros do pneu e da gravação que a máquina necessita de receber para poder realizar o processo. Será vista detalhadamente na secção 4.3 - Descrição comportamental do funcionamento da máquina.

[12] Esta decisão foi tomada porque senão teria de se desenvolver uma base de dados relativamente complexa para testar o código feito.

Para fazer face a isto, decidiu-se que duas das saídas digitais trocadas entre o PLC de linha e o PLC da máquina identificariam o estado (Ligado ou Desligado) dos dois transportadores das estações de entrada e saída máquina, e que duas das entradas digitais identificariam o estado dos transportadores de entrada e saída da fábrica. Esta situação foi descrita na subsecção anterior.

Segue-se uma análise das várias situações de paragem da máquina com necessidade de reinício, e maneira como este protocolo se adapta a elas. Esta análise faz referência às três situações representadas na figura 4.9.

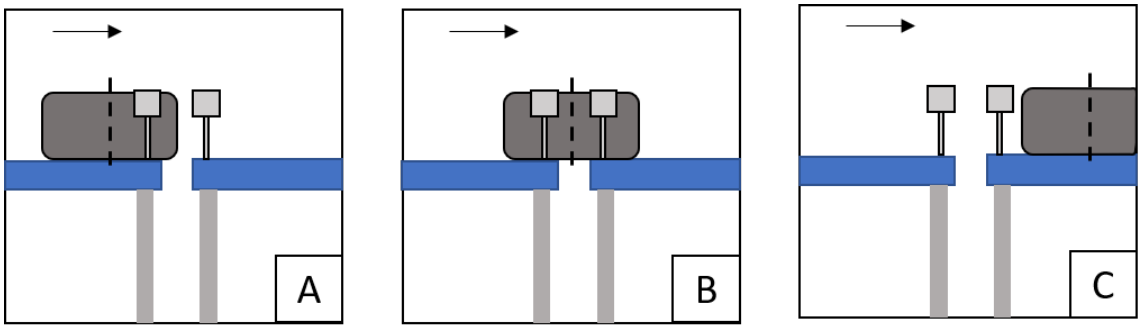


Figura 4.9: Esquema das etapas de transferência de um pneu.

Atentando à situação de transferência de um pneu do transportador da fábrica para a estação de entrada:

- Paragem antes de o pneu ser detetado pelo sensor do transportador da estação de entrada (situação A):

  O pneu não é considerado como dentro da máquina, continua disponível na tabela de entrada e não está em cima do transportador da estação de entrada. O transportador da fábrica continua no estado "pronto a dar" e assim, após reinício com sucesso pode-se proceder à retoma da transferência.

- Paragem quando o pneu está a ser detetado pelos sensores de ambos os transportadores (situação B):

  O pneu é considerado como dentro da máquina, continua disponível na tabela de entrada e está em cima do transportador da estação de entrada. Durante a rotina de reinicialização, são feitas movimentações no sentido da entrada para a saída, pelo que, ao verificar que o transportador da estação de entrada se está a mexer, o PLC de linha irá energizar novamente o seu transportador, e concluir a transferência.

  Adicionalmente, durante a rotina de reinicialização é feita uma verificação do facto de o pneu na estação de entrada ainda estar presente na tabela de entrada. Em caso positivo, ele será eliminado desta tabela.

- Paragem depois de o pneu deixar de ser detetado pelo sensor do transportador da estação de entrada (situação C):

  Esta é a situação normal. O pneu não está no transportador da fábrica, já foi removido da tabela de entrada e está totalmente dentro da máquina.

Atentando agora à transferência de um pneu da estação de saída para o transportador da fábrica:

- Paragem antes de o pneu ser detetado pelo sensor do transportador da fábrica (situação A):

  O pneu não é considerado como fora da máquina, continua disponível na tabela de saída e não está em cima do transportador da fábrica. O transportador da máquina continua no estado "pronto a dar" e assim, durante a etapa de reinicialização retoma-se a transferência.

- Paragem quando o pneu está a ser detetado pelos sensores de ambos os transportadores (situação B):

  O pneu é considerado como dentro da máquina, continua disponível na tabela de saída e está em cima do transportador da fábrica. Durante a rotina de inicialização, tenta-se escrever novamente na tabela de saída sobre o pneu na estação de saída e volta-se ao estado de "pronto a dar". Visto que o transportador da fábrica se mantém "pronto a receber", então a transferência é concluída.

- Paragem depois de o pneu deixar de ser detetado pelo sensor do transportador da estação de saída (situação C):

  Esta é a situação normal. O pneu está no transportador da fábrica, já foi removido da tabela de saída e está totalmente fora da máquina.

Durante a programação adotou-se a seguinte regra: os pneus não precisam de ser retirados manualmente da máquina pelo operador, porque a máquina os consegue transferir autonomamente. Caso, por algum motivo pontual, a regra seja desrespeitada e o pneu for retirado à máquina, um erro será assinalado e a máquina terá de ser reinicializada. A rotina de reinicialização será descrita na secção 4.3.3 - *Grafcet* 2 - Rotina de reinicialização.

Adianta-se que em qualquer altura do funcionamento, o operador terá disponível na HMI quais os pneus que a máquina considera que estão em cada estação, permitindo compreender o estado do processo. Como nota final do protocolo apresentado, sugere-se que se a qualquer altura se adotar um outro sistema de fluxo de informação em vez do servidor SQL, este protocolo poderá ser utilizado na mesma.

A comunicação com o servidor SQL é feita através da programação em VBA, implementada num dos ecrãs do SCADA que está constantemente aberto. Alternativamente, o próprio PLC também poderia fazer pedidos ao servidor SQL desde que este aceitasse comunicação por *sockets* TCP/IP, o que seria vantajoso do ponto de vista de fazer com que o IPC deixasse de ser um intermediário de informação. Deve-se notar que a parte complexa da interface com o servidor SQL não é a programação em VBA, mas sim a criação do protocolo explicado no início da secção.

Quando o PLC precisa de fazer um pedido ao servidor, fá-lo através de um pedido ao SCADA, que por sua vez faz o pedido ao servidor. A resposta do servidor percorre o mesmo caminho até ao PLC. Nos parágrafos seguintes, será descrita a metodologia adotada para realizar essa comunicação do ponto de vista prático.

A utilização do VBA para este efeito, permite recorrer à *Dynamic Link Library* (DLL) denominada *ActiveX Data Object* (ADO). Uma DLL é um conjunto de funcionalidades disponíveis para serem utilizadas por vários programas ao mesmo tempo. Estas livrarias promovem a reutilização de código e utilização eficiente da memória [49], na medida em que não são refeitas em cada programa que as utiliza .

A utilização desta DLL permite comunicar com bases de dados, independentemente do seu tipo (pode ser um servidor MySQL, Oracle, Microsoft SQL server, entre outros) desde que a conexão seja feita. Adicionalmente, permite criar objetos do tipo ADODB.Connection, ADODB.Recordset e ADODB.Command que representam uma conexão à base de dados, um conjunto de dados e um comando a enviar ao servidor, respetivamente. O ADO constitui uma interface entre a linguagem programada (Visual Basic) e o *Object Linking and Embedding - Data Base* (OLEDB) [50].

Por sua vez, o OLEDB é uma *Application Programming Interface* (API) - um conjunto de funções e rotinas da Microsoft que realmente executa as tarefas de aceder a dados de uma base, recorrendo à plataforma *Component Object Model* (COM), também da Microsoft [51]. O COM é utilizado para definir a criação de objetos que servem para comunicação entre processos, independentemente da linguagem de programação [52]. Os objetos *ActiveX Data Object - Data Base* (ADODB) referidos no parágrafo anterior são fruto do COM.

A Microsoft criou o OLEDB como um sucessor do *Open Database Connectivity* (ODBC), porque extende as suas funcionalidades a bases de dados não relacionais. O ODBC (não aqui utilizado) é igualmente uma API para aceder a bases de dados [51]. Existe, no entanto, um programa do Windows chamado "ODBC Data Sources" (disponível nos computadores com Windows), que é utilizado e é compatível com o OLEDB.

Este programa serve para configurar conexões a bases de dados, através da utilização de um *driver* da respetiva base de dados. Deste modo, é criada uma *data Source* (ou fonte de dados) que contém informações sobre a conexão, guardada no próprio computador. Assim, é possível que o programa desenvolvido em VBA aponte para uma dada fonte de dados, em vez de guardar os credenciais de autenticação na base de dados dentro do próprio código.

Desta forma, se o cliente pretender a qualquer altura modificar as credenciais da base de dados, não precisa de fazer alterações ao código em VBA. Basta assim que atualize as credenciais de ligação ao servidor no programa ODBC *Data Sources*.

No que toca ao servidor SQL, existem dois tipos de autenticação: o primeiro é autenticação pelo próprio Windows, e requer que o IPC e o servidor SQL estejam no mesmo domínio. Um domínio é um grupo de computadores em rede devidamente identificados, gerido por um servidor que tem a função de autenticar cada um dos participantes. O segundo tipo de autenticação é por nome de utilizador e palavra-chave que, apesar de tornar o servidor mais vulnerável, foi utilizado nos testes feitos.

Do ponto de vista de programação em si, o programador apenas precisa de conhecer os objetos ADODB.Connection, ADODB.Command e ADODB.Recordset, podendo-se abstrair dos conceitos acabados de apresentar. Apesar de prescindíveis, não seria razoável que estes conceitos não fossem aqui abordados porque é importante ter uma noção das tecnologias sobre as quais se trabalha.

O objeto ADODB.Connection é necessário para criar uma ligação a uma base de dados, e é preciso parametrizar os dados de uma conexão. O objeto ADODB.Command serve para executar comandos SQL, por exemplo, pedir para ler uma linha de uma tabela, que será armazenada num objeto do tipo ADODB.Recordset. Todos estes objetos contêm métodos e propriedades que dão ao utilizador imensa versatilidade na programação.

Voltando à programação em VBA, recorre-se aos três objetos referidos para efetuar todas as tarefas de comunicação com bases de dados necessárias para esta aplicação. Pode-se resumir as funções a implementar numa lista:

- Leitura da primeira linha da tabela de entrada;
- Eliminação da primeira linha da tabela de entrada com um dado ID do pneu;
- Leitura da primeira linha da tabela de saída;
- Eliminação da primeira linha da tabela de saída;
- Escrita da primeira linha da tabela de saída.

Durante a receção/envio de um pneu, o PLC faz pedidos ao SCADA para correr as várias funções referidas.

Os pedidos são feitos através da escrita de variáveis booleanas do PLC, que estão ciclicamente a ser monitorizadas pelo SCADA (por exemplo, a cada 10 s). Quando uma operação é completa, o SCADA altera outro booleano com o significado "Operação Completa" que é devidamente identificado pelo PLC. Em caso de operação não completa, o programa voltará a tentar.

A conexão com o servidor deve estar sempre aberta. Em caso de falha na comunicação, o programa em VBA tentará reconectar-se a cada 10 s, desde que exista um pedido de comunicação feito pelo PLC. Adicionalmente, o operador será devidamente alertado e o ciclo de processo automático será suspenso.

Durante a etapa de receção de um pneu, os comandos de leitura e eliminação da linha da tabela de entrada são enviados um a um pelo PLC, visto que acontecem em momentos diferentes. Já durante o envio de pneu, o PLC apenas faz o pedido de escrita da informação de pneu na tabela de saída. Quando isto acontece, o programa executa em primeiro lugar uma leitura da linha para verificar se a tabela está vazia: caso ainda esteja lá a informação do pneu anterior, o programa irá esperar até que este desapareça.

Tanto após a eliminação de um pneu na tabela de entrada, como após a escrita de um pneu na tabela de saída, o programa executa uma leitura da primeira linha dessa tabela a confirmar se a ordem realmente foi executada com sucesso. Em caso negativo, o PLC tentará eliminar/escrever de novo. Caso seja detetada a presença de um pneu inesperado na tabela, é lançado um erro e o ciclo é suspenso necessitando que o operador resolva o conflito.

Decidiu-se ainda implementar uma funcionalidade de comando do operador para consultar ou eliminar um dado pneu nas tabelas saída/entrada, apenas disponível em comando manual. Isto dá flexibilidade ao operador para controlar o processo e resolver problemas.

Do ponto de vista de implementação prática de software, o FactoryTalk SE contém um editor de VB. A criação de código é feita associada a cada ecrã, pelo que se utilizou um ecrã que está sempre aberto (o painel de cabeçalho). O código é constituído por vários procedimentos, cada um com um objetivo bem definido.

Existem procedimentos de objetos despoletados por eventos. Por exemplo, neste caso foi utilizada uma caixa numérica (*Numeric Display*) invisível, ao qual está associada a variável interna do sistema *System/Seconds* cujo valor é o número de segundos passados desde o minuto anterior. O objeto neste caso é a caixa numérica e o evento é a mudança do valor da variável alocada.

Quando esta caixa de texto toma um valor novo, é corrido um procedimento que verifica se o número de segundos passado é múltiplo de dez. Em caso afirmativo, é despoletada a lógica acima descrita em que são verificados os pedidos do PLC para comunicação com o servidor SQL, chamando outros procedimentos necessários em função daquilo que é comandado pelo PLC.

Assim fica completa a explicação de como é programada a interação da máquina com o servidor SQL da fábrica.

4.2.8 Interface com cilindros pneumáticos

Os cilindros pneumáticos são componentes comuns em máquinas nesta indústria, tornando-se altamente útil a criação de uma AoI para estes componentes. Desta forma, a programação de um novo cilindro é tão simples quanto a criação e ligação das variáveis do programa. Esta AoI foi fornecida pela empresa e sofreu apenas pequenas alterações.

Esta instrução permite o controlo de um cilindro pneumático de duplo efeito sensorizado nos dois extremos do seu curso. O seu controlo pode ser feito através do PLC (durante o ciclo automático) ou da HMI (durante o controlo manual). Adicionalmente, são fornecidas à AoI variáveis adicionais tais como *interlock* e *emergenciaOK*, para que os solenóides das válvulas não possam ser atuados (não querendo com isto tentar implementar uma emergência por software).

O controlo é implementado da seguinte maneira: existem duas variáveis booleanas para comandar o movimento do cilindro em ambas as direções. O sinal dos sensores nos seus extremos impede que o solenóide continue a ser energizado, quando um extremo já foi alcançado. Desta forma, é implementado o controlo dos cilindros que não carece de nenhuma explicação adicional.

4.2.9 Interface com eixos servomotorizados

Um eixo é genericamente formado pelo servomotor, uma transmissão, um sensor em cada extremo do curso da carga e um sensor de *homing*. Os seus sensores fim de curso são utilizados para forçar a paragem do motor e impedir futuro acionamento no mesmo sentido. O sensor de *homing* é utilizado para o controlador ter uma referência absoluta de posição.

Tal como para os cilindros pneumáticos, a empresa forneceu uma AoI de controlo, onde já haviam sido programadas as várias instruções de movimento necessárias para

o processo, abordagem da situação em que o eixo alcança os fins de curso, variáveis de *status* e controlo do próprio eixo.

O *homing* é uma rotina que o servomotor realiza para encontrar a sua posição absoluta, e pode ser feito de várias maneiras, conforme parametrizado. Existem dois modos para fazer o *homing*: ativo e passivo.

No modo passivo, o eixo fica imóvel à espera de que a sua rotação seja feita externamente e uma marca de referência do codificador seja detetada. Este modo é principalmente utilizado em eixos apenas de *feedback*, por exemplo, para fazer o *home* de um possível eixo virtual da mesa através do sinal do codificador externo, vindo do *splitter*.

Dentro do modo ativo existem quatro tipos de *homing*:

- Imediato;
- Com sensor;
- Com marca do codificador;
- Com sensor e marca do codificador.

O tipo imediato significa que à posição onde o servomotor se encontra será referenciada uma posição escolhida. Para realizar este tipo de *homing*, o operador teria que:

1. Ter uma pequena régua colocada ao lado do eixo a indicar a posição de "home";
2. Mover o eixo manualmente para essa posição;
3. Efetuar o comando de *home* através da HMI.

Este processo tornar-se-ia temporalmente dispendioso e estar-se-ia a depender da perícia do operador para que a rotina de *homing* fosse feita. Dado que se trata de eixos com difícil acesso, não se utilizará este tipo de *homing*.

O *homing* com sensor é feito através de uma rotina disponível de fábrica. Esta rotina faz com que o eixo percorra o seu curso até o sensor o detetar, seguido de uma inversão da marcha na direção oposta até que o sensor o volte a detetar. Feito isto, é feita uma nova aproximação ao sensor, parando na posição de *home*. A razão para a rotina ser feita com várias passagens pelo sensor é devido ao facto de a carga não ser detetada apenas numa posição, mas sim numa gama de posições, tornando-se este método aplicável à utilização de sensores de de proximidade.

O *homing* através de marca de codificador é apenas para eixos univolta, nos quais a marca de um codificador só se dá uma vez em todo o seu curso.

Finalmente, o *homing* com marca do codificador e sensor é o mais preciso de todos, e consiste na execução de uma rotina em que o eixo percorre todo o seu curso à procura de encontrar a posição do sensor, seguindo-se um retorno à procura da posição da marca do codificador. Sendo este último o método mais preciso de todos, requer que a marca do codificador esteja disponível e que o sensor de *homing* seja preferencialmente de contacto mecânico.

Dito isto, opta-se pelo *homing* ativo, apenas com sensor, porque se prevê que será utilizado um sensor de proximidade. Caso fosse utilizado um sensor de contacto mecânico a meio do curso, o movimento cíclico do eixo iria levar a um desgaste precoce, que é indesejável.

Do ponto de vista de controlo dos servomotores, é ainda necessário ajustar os seus parâmetros de controlo à aplicação real, constituindo assim um trabalho futuro.

O PLC dá instruções aos eixos através da escrita dos parâmetros de entrada desta AoI. Em troca, o *feedback* recebido dos servomotores é devidamente organizado, e devolvido como um parâmetro de saída. Desta forma, a integração no ciclo de processo automático pode ser feita recorrendo apenas a estas variáveis, sem ter de reprogramar complexas instruções de movimento.

4.3 Descrição comportamental do funcionamento da máquina

Esta secção tem como objetivo expor o comportamento programado através de *grafcets* que serão devidamente apresentados ao longo dos parágrafos. Porém, para maximizar a compreensão do texto é necessário introduzir duas variáveis-chave do processo: a receita do pneu e o seu *status*.

As variáveis abaixo apresentadas foram implementadas sobre a forma de UDTs e servem para guardar informação sobre o processo e estado dos pneus.

O *status* do pneu é uma variável que existe em cada estação, que contém informação sobre o pneu. O valor dos membros desta variável está ligado ao pneu que está fisicamente em cada estação, pelo que ao transferir o pneu entre estações é necessário transferir também as respetivas *status*. Note-se que existem três instâncias desta variável, uma para cada estação.

Esta variável tem dois conjuntos de membros: aqueles que dizem respeito ao estado do processamento do pneu e aqueles que dizem respeito às dimensões do pneu e tipo de gravação que deve ser feita. Os membros que constituem a variável são apresentados na tabela 4.6, bem como o seu significado. Durante a análise dos *grafcets*, qualquer referência a um membro da variável *status* é feita através da notação *status.<membro>*.

Tabela 4.6: Membros da variável *status* do pneu.

Nome	Tipo de variável	Descrição
Completo	Booleano	*True*: Operação de gravação completa
Abortado	Booleano	*True*: Pneu abortado, não gravado
ID	Inteiro de 32 bit	Chave primária que identifica cada pneu
Existe	Booleano	*True*: Um pneu associado a esta estação existe
Receita	UDT - Receita de pneu	Contém todas as informações necessárias ao processo

À medida que o ciclo vai sendo percorrido, o valor de cada um destes membros vai sendo atualizado em função do resultado do processo a acontecer. A máquina obtém a receita e ID do pneu através de comunicação com o servidor SQL. Na tabela 4.7 estão descritos os vários membros da UDT *Receita*.

Tabela 4.7: Membros da variável receita do pneu.

Nome	Tipo de variável	Descrição
DiametroExterno	Real	Diâmetro externo em mm do pneu em questão
DiametroInterno	Real	Diâmetro interno em mm do pneu em questão
Largura	Real	Largura em mm do pneu em questão
AngMarcRef	Real	Ângulo em graus entre a posição a gravar e a referência
DiametroRef	Real	Diâmetro em mm a que se encontra a referência
NomeFicheiro	String-12	Nome do ficheiro no LASER a gravar no pneu
DiametroMarc	Real	Diâmetro em mm da posição onde se deve fazer a gravar
TipoRef	Inteiro de 32 bit	Tipo de referência que a câmara deve reconhecer
RacioRef	Inteiro de 32 bit	Razão largura altura da referência

A finalidade desta máquina é gravar a parede lateral dos pneus automaticamente, ou seja, sem que exista um operador a controlar constantemente a máquina. Para poder atingir este nível de autonomia, é necessário implementar dois modos de funcionamento da máquina: modo manual e automático.

A troca entre modo manual e automático é determinada pela posição do seletor disponível montado no painel da HMI (descrito na secção 4.2.1 - Elementos de comando da HMI). Note-se que a troca entre modos pode ser feita a qualquer altura, independentemente de se estar a meio de um ciclo ou não[13].

O modo de funcionamento manual serve, como o nome indica, para comando manual da máquina. Este modo é necessário para fazer operações de manutenção, configuração, diagnóstico e teste. O controlo manual por parte do operador é feito através de comandos na HMI, e aplica-se aos seguintes componentes do sistema:

- Transportadores de rolos;
- Cilindros pneumáticos;
- Eixos servomotorizados;
- Operações com o servidor SQL.

O comando manual de componentes móveis é possível enquanto a botoneira *Move* no painel da HMI estiver a ser pressionada, permitindo fazer controlo sensitivo. Desta forma, o operador terá maior controlo sobre o movimento.

Passa-se à apresentação dos *grafcets* criados para descrever o comportamento da máquina. Existem seis grafos, que interagem hierarquicamente e serão apresentados na ordem abaixo. Essa ordem foi escolhida para maximizar a perceção da leitura.

1. *Grafcet* 1 - Controlo do modo automático;
2. *Grafcet* 0 - Controlo central;
3. *Grafcet* 2 - Rotina de reinicialização;

[13]Um ciclo automático interrompido pela entrada no modo manual não poderá ser retomado: terá de se fazer uma expulsão dos pneus no interior da máquina (rotina de reinicialização) e o ciclo será recomeçado do início.

4. *Grafcet* 3 - Ciclo automático da estação de entrada;
5. *Grafcet* 5 - Ciclo automático da estação de saída;
6. *Grafcet* 4 - Ciclo automático da estação central.

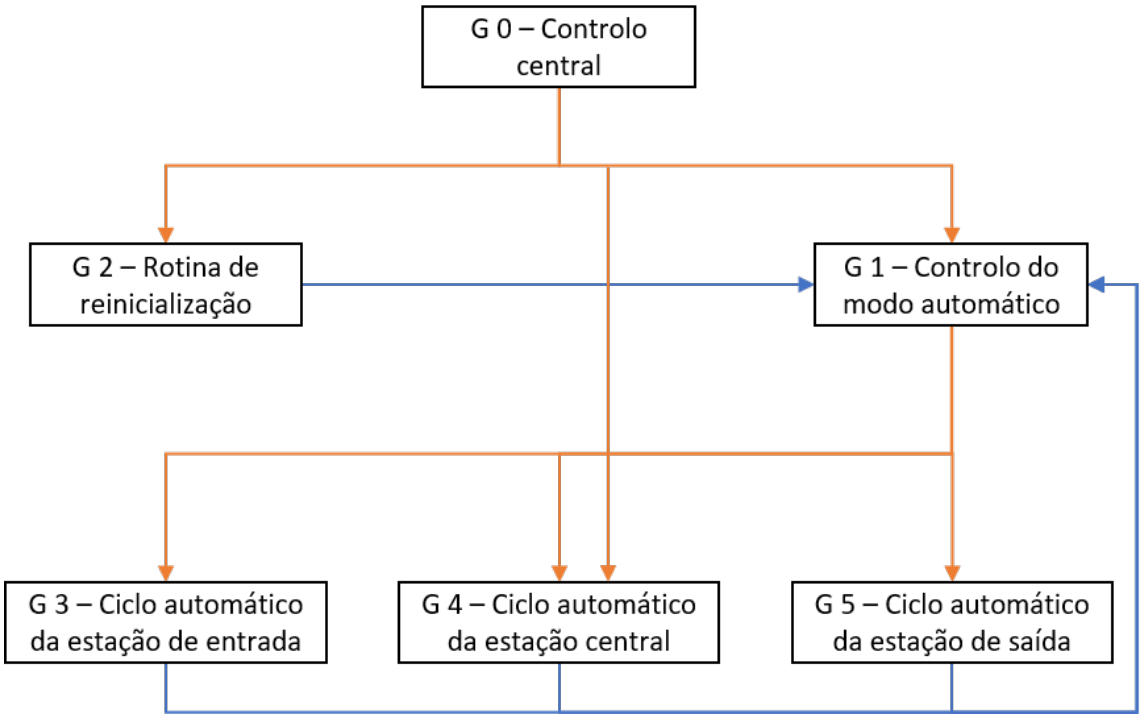


Figura 4.10: Diagrama hierárquico de interação entre *grafcets*. Linhas azuis: *Feedback*. Linhas cor-de-laranja: ordens.

A interação entre *grafcets* está representada na figura 4.10, onde são diferenciadas ordens e *feedback*. Uma ordem (linhas cor-de-laranja) significa que um grafo comanda o outro na direção da seta através do congelamento ou forçagem a uma etapa. *Feedback* (linhas azuis) significa que são utilizadas variáveis controladas pelo grafo de onde parte a seta para provocar transições nos grafos para onde a seta aponta.

4.3.1 *Grafcet* 1 - Controlo do modo automático

O modo automático é aquele em que é possível colocar a máquina a funcionar autonomamente e pode ser traduzido através do *grafcet* 1 representado na figura 4.11. Dentro deste modo existem quatro etapas de funcionamento:

1. Não inicializado;
2. Em suspensão;
3. A suspender;
4. Em processo.

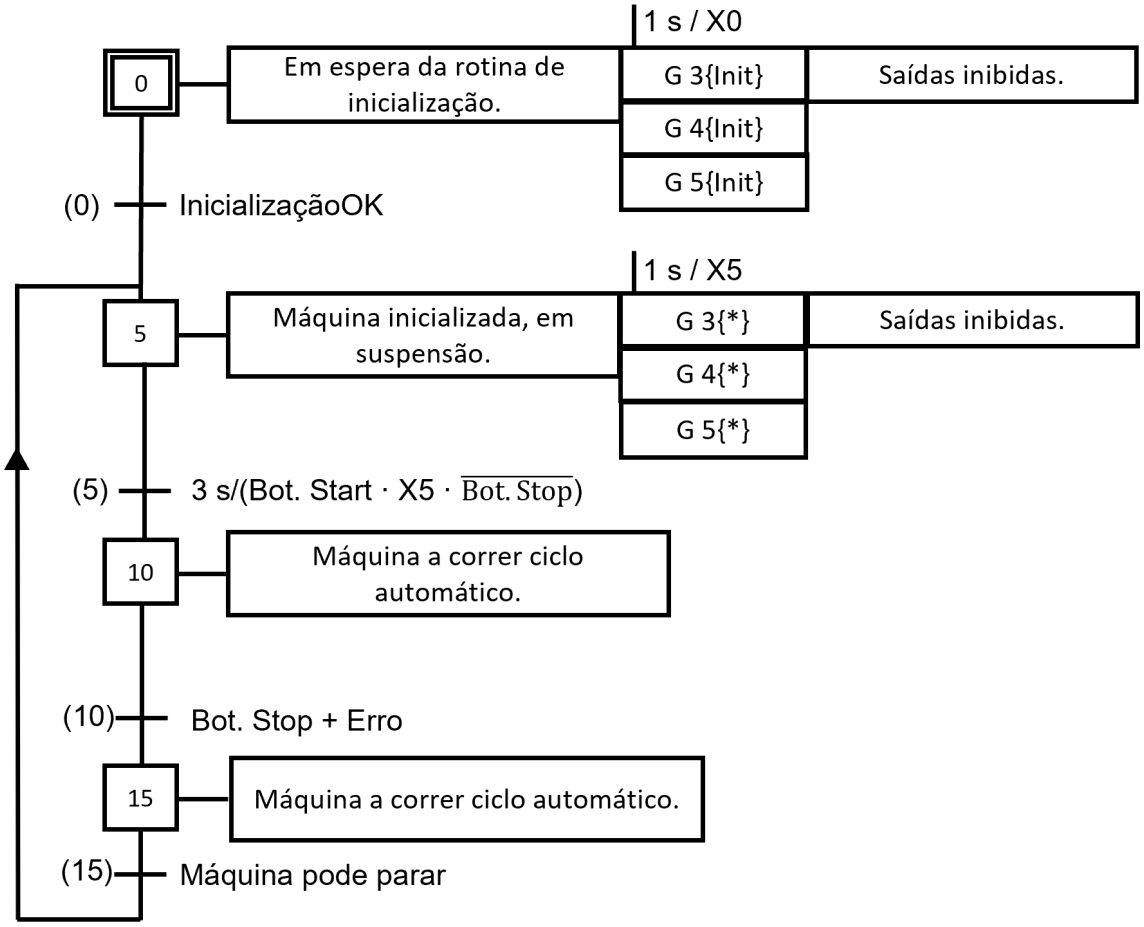


Figura 4.11: *Grafcet* 1 - Controlo do modo automático.

Para poder entrar em processo, é preciso que tenha sido previamente realizada uma rotina chamada rotina de reinicialização (responsável pela variável *InicializaçãoOK*), que garante que não existe nenhum pneu nas estações da máquina. Após isto, é possível navegar entre as três etapas através das botoneiras *start*[14] e *stop*. Este *grafcet* é responsável por controlar os *grafcets* 3, 4 e 5 que correspondem aos ciclos automáticos das três estações da máquina: entrada, central e saída.

Na mesma figura, pode-se verificar que as ações de congelamento e reinicialização dos ciclos automáticos das estações são programadas com um atraso ao ligar relativamente à inibição das saídas. Este desfasamento entre ações garante que a lógica do ciclo apenas deixa de ser executada 1 s após as saídas terem sido inibidas, para que um possível escorregamento do pneu devido à sua inércia seja detetado. Caso contrário, poderia acontecer que o pneu tivesse feito uma transição importante, por exemplo uma passagem entre o transportador da estação de saída da máquina e da fábrica, que não tivesse sido detetada.

A etapa 15 serve para acomodar situações em que se pretende fazer um pedido de suspensão à máquina. Existem etapas do funcionamento automático que não podem ser suspensas (por exemplo a gravação a LASER e a rotina da câmara), pelo que se

[14] A botoneira *start* tem de ser pressionada durante mais de três segundos para sair da suspensão, como forma de garantir que não foi pressionada por engano.

a botoneira *stop* for pressionada ou for detetado um erro, dar-se-á a transição 10. A transição 15 dá-se apenas quando o ciclo automático da estação central alcançar uma etapa na qual o ciclo possa ser suspenso.

4.3.2 *Grafcet* 0 - Controlo central

O controlo do *grafcet* 1 é feito através do *grafcet* 0, responsável pelo controlo central (também chamado de *grafcet* mestre). Este grafo (disponível na figura 4.12) de apenas uma etapa serve para execução de ações de controlo dos restantes *grafcets*, que devem estar sempre disponíveis.

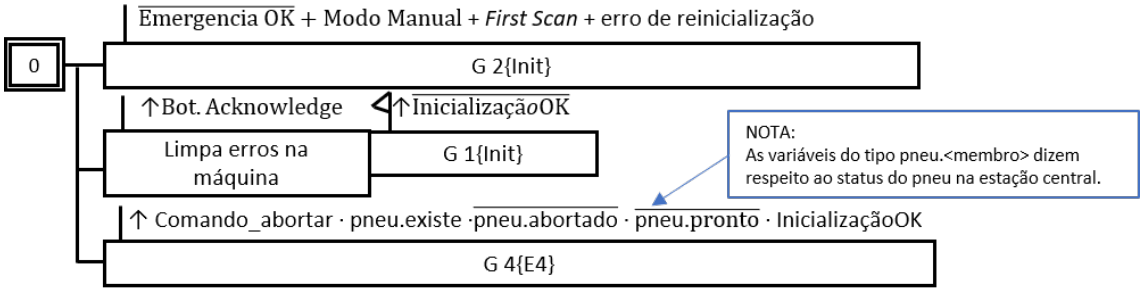


Figura 4.12: *Grafcet* 0 - Controlo central.

A primeira ação do *grafcet* 0 serve para determinar que a rotina de reinicialização precisa de retornar à etapa inicial e impede que esta se desenvolva, sempre que uma de quatro condições se verificam:

- Existir uma situação de emergência:

  Por exemplo, se alguém abrir uma porta de correr da máquina significa que o determinismo dos acontecimentos num ciclo poderá ter sido comprometido.

  Poder-se-ia arguir que não é necessário ser tão drástico: bastava reiniciar a respetiva estação. No entanto, caso um dos portões estivesse aberto, o operador estaria ao alcance de comprometer também os ciclos das outras estações através de um sinal falso num sensor ou movimento do pneu;

- Entrada em modo manual:

  Durante o funcionamento em modo manual, podem ser executados comandos de movimento dos componentes da máquina que levariam os pneus para posições desconhecidas;

- *First Scan*:

  Este bit toma o valor *True* sempre que uma tarefa é corrida pela primeira vez depois de ter sido inibida, pelo que existe um *First Scan* para cada tarefa e foi considerado por dois motivos. O primeiro motivo é em caso de falha de energia da máquina e independentemente da existência ou não de uma UPS (Uninterruptible Power Supply), ao voltar a ligar, a máquina não poderá retomar o ciclo onde estava antes de desligar.

O segundo motivo está relacionado com a passagem do controlador do *Program Mode* para o *Run Mode*, por exemplo, quando há uma descarga de um programa do Studio 5000 para o PLC. O facto de o controlador estar em *Program Mode* significa que a lógica não está a ser executada [53], comprometendo o determinismo do ciclo;

- Erro de reinicialização:

  Se durante a rotina de reinicialização for detetado na máquina um pneu inesperado, ocorre este erro. Neste caso, o operador deve reconhecer o erro e proceder à remoção do pneu da máquina. Esta é a única situação prevista em que o pneu pode ser removido manualmente da máquina, porque não são conhecidas as dimensões do pneu em questão.

Este grafo é ainda responsável por limpar erros e levar o *grafcet* 1 de volta à sua etapa inicial, sempre que a rotina de reinicialização necessitar de ser executada do início.

A quarta ação diz respeito ao comando de abortagem: se um pneu estiver na estação central, ainda não tiver sido abortado, a marcação ainda não tiver sido executada, não for necessária uma reinicialização e o operador der um comando para abortar através da HMI, a estação central irá proceder à abortagem do pneu. A rotina de abortagem faz parte do ciclo automático da estação central.

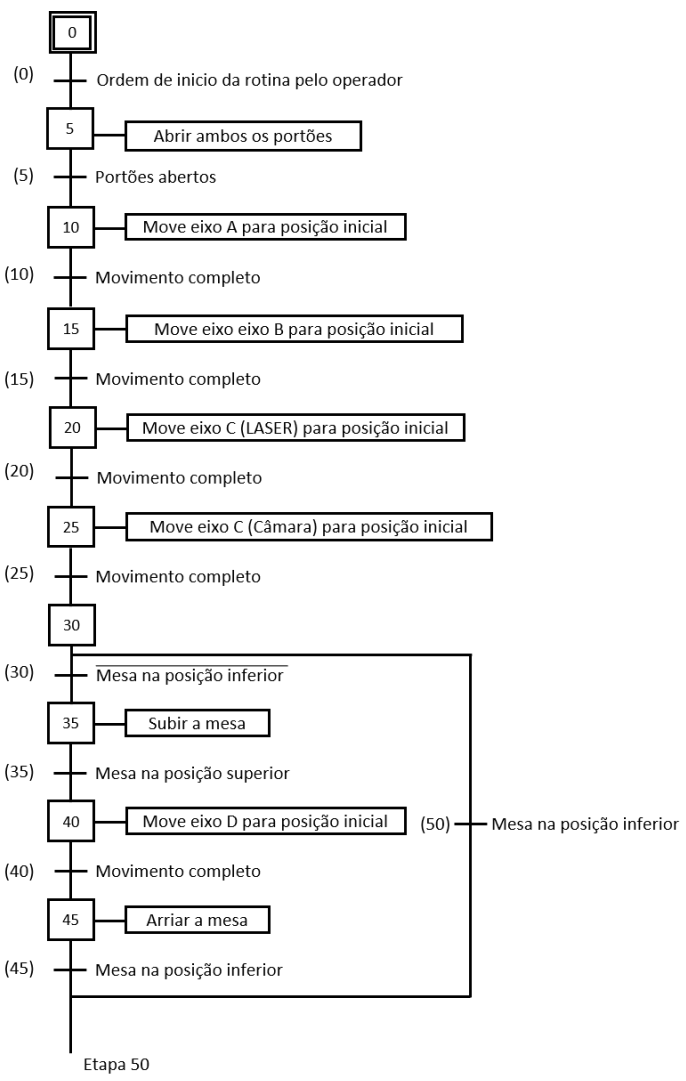
4.3.3 *Grafcet* 2 - Rotina de reinicialização

O *grafcet* 2, que é controlado pelo *grafcet* 0 trata-se do ciclo de reinicialização. O início desta rotina (figura 4.13a) é dispoletado por uma ordem de confirmação do operador através de um botão programado num ecrã do SCADA. A rotina começa com a abertura dos portões e colocação dos eixos A, B, C_{LASER}, C_{camara} e D[15] na sua posição de início de ciclo (que evitará colisões com pneus a circular no transportador).

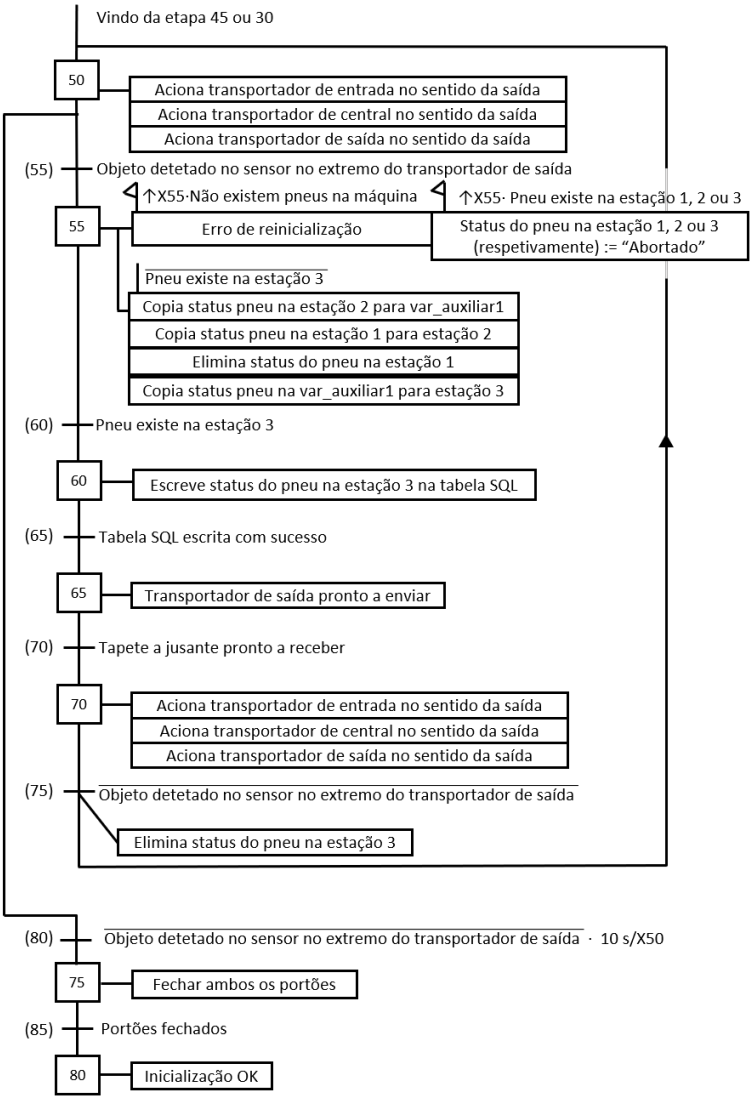
Feito isto, os três transportadores são colocados em movimento para que qualquer pneu que esteja na máquina seja transportado até ser detetado pelo sensor no extremo final do transportador de saída (figura 4.13b). Quando este sensor deteta algum pneu, ele é passado para a estação seguinte com o respetivo *handshake* com o servidor SQL e o PLC de linha. O comportamento descrito neste parágrafo repete-se ciclicamente até que passem 10 s (parâmetro temporal a ajustar) desde que os transportadores começam a funcionar e não seja detetado qualquer pneu, assumindo que a máquina está vazia.

Não detetando nenhum pneu no interior, fecham-se ambos os portões e chega-se à etapa final - uma etapa poço com uma ação não memorizada. O facto de a ação ser não memorizada significa que ao sair desta etapa, o valor da variável *InicializaçãoOK* é *False*. Como se pode verificar das secções anteriores, esta variável é utilizada como condição necessária para que as estações estejam em processo, correndo o seu ciclo automático.

[15] O eixo D apenas é movido se a mesa não estiver na sua posição inferior. Caso contrário assume-se que já se encontra na posição de início de ciclo.



(a) Parte 1 de 2.



(b) Parte 2 de 2.

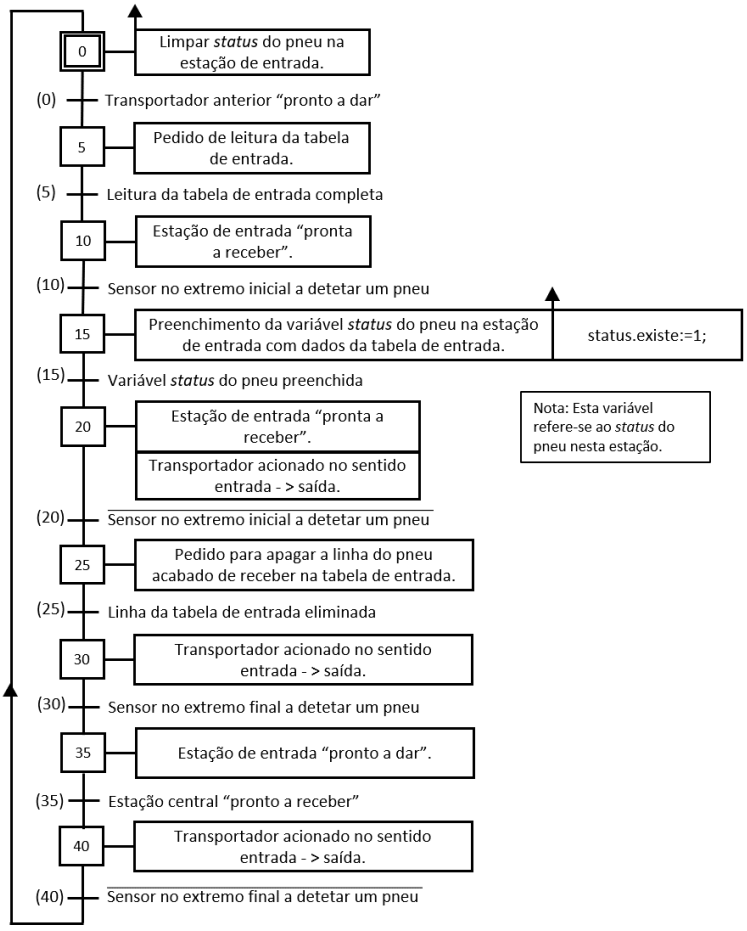
Figura 4.13: *Grafcet* 2 - Rotina de reinicialização.

4.3.4 *Grafcet* 3 e 5 - Ciclos automáticos das estações de entrada e saída

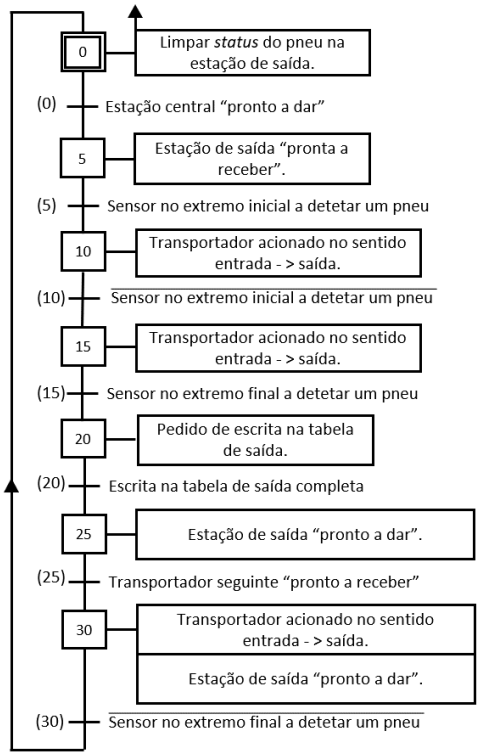
O ciclo em modo automático da estação de entrada (figura 4.14a) começa com a limpeza da variável *status* do pneu, que vem preenchida do ciclo anterior e deve ser apagada visto que o transportador nesta altura não deve ter nenhum pneu. Após isto, é seguido o protocolo de receção de um pneu de um transportador que passa pela leitura da tabela de entrada do servidor SQL, já descrito na secção 4.2.7 - Interface com o servidor SQL. Ao ver que o transportador anterior está pronto para enviar um pneu, a estação faz um pedido para ler a primeira linha da tabela de entrada.

Feita a leitura, a estação entra no modo "pronto a receber", permitindo o transportador a montante da máquina saber que pode acionar o seu transportador. Ao

detetar o pneu à entrada, é preenchida a variável *status* do pneu na estação central com as informações obtidas do servidor SQL e procede-se ao acionamento do transportador desta estação. No momento em que o pneu fica completamente dentro da estação, é feito um pedido ao servidor SQL para apagar a linha que diz respeito a este pneu.



(a) *Grafcet* 3 - Ciclo automático da estação de entrada.



(b) *Grafcet* 3 - Ciclo automático da estação de saída.

Figura 4.14

Findo isto, o pneu é transportado de uma ponta à outra do transportador até atingir o sensor no seu extremo final. Nessa altura, entra no modo "pronto a dar", ficando à espera de que a estação central fique no modo "pronto a receber". Quando isto acontece, o transportador é novamente acionado até que o pneu deixe de ser sentido pelo seu sensor no extremo final, altura em que se assume que o pneu deixa de estar nesta estação.

Note-se que entre as estações entrada/central e central/saída não é preciso fazer qualquer troca de informação com o servidor SQL. A cópia da variável *status* do pneu da estação de entrada para o *status* do pneu da estação central é feita numa ação do ciclo da estação central.

O ciclo da estação de saída (figura 4.14b) apenas difere do da de entrada na

medida em que faz a escrita de uma linha da tabela no servidor SQL, ao passo que no anterior se fazia leitura e eliminação da linha da tabela.

Quanto a estação central se encontra "pronta a dar", a estação de saída entra no modo "pronto a receber" e procede-se à transferência do pneu. O ciclo continua até à etapa 20, altura em que se pretende escrever na tabela de saída do servidor SQL.

Se a escrita for feita com sucesso, então esta estação entra no modo "pronto a dar" e, quando deteta um "pronto a receber" por parte do transportador da fábrica a jusante desta estação, procede ao acionamento do transportador. Deixando de detetar o pneu, volta à etapa inicial na qual limpa a variável *status* do pneu na estação de saída.

4.3.5 *Grafcet* 4 - Ciclo automático da estação central

O ciclo da estação central é o mais importante para o processo e também o mais extenso: está representado em duas partes na figura 4.15. Para facilitar a sua perceção e encapsular partes específicas do ciclo, foram criadas quatro macroetapas que dizem respeito a:

1. Verificação das dimensões do pneu;
2. Determinação da posição da referência do pneu (etapa da câmara);
3. Rotina de gravação a LASER;
4. Rotina de abortagem do pneu.

O ciclo começa com a transferência do pneu da estação de entrada para a estação central de acordo com o que já foi explicado. Na etapa 15, é feita a cópia da variável *status* do pneu da estação de entrada para a estação central, garantindo o fluxo de informação.

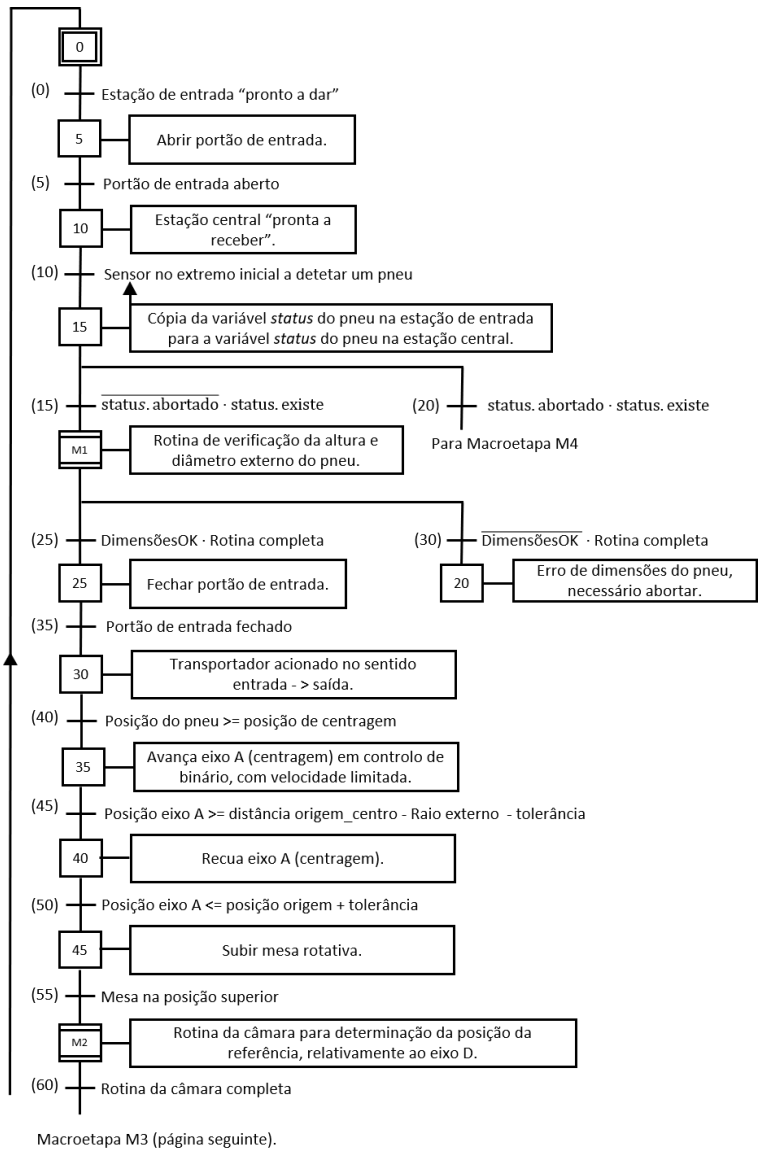
Ao receber, é possível que o pneu em questão já tenha sido identificado como abortado, pelo que pode imediatamente ser expelido desta estação. Por exemplo, imagine-se que se quer remover todos os pneus da linha sem que sejam gravados, mas têm que passar pela máquina porque não há outro caminho. Assim, a divergência através da transição 20 permite saltar imediatamente para a subrotina de abortagem do pneu.

Caso o pneu não seja abortado, é feita a rotina de verificação de largura e diâmetro externo do pneu (macroetapa M1 - figura 4.16). A medição destas grandezas é feita com a utilização de uma barreira de medição e um codificador acoplado a um dos rolos do transportador. A rotina consiste na passagem do pneu pelos feixes da barreira, aquisição das grandezas referidas e comparação com os valores obtidos na tabela de entrada. Caso a largura ou o diâmetro externo de um dado pneu estejam fora de um certo intervalo em torno dos respetivos valores de referência, é lançado um erro para o operador, suspendendo o ciclo automático.

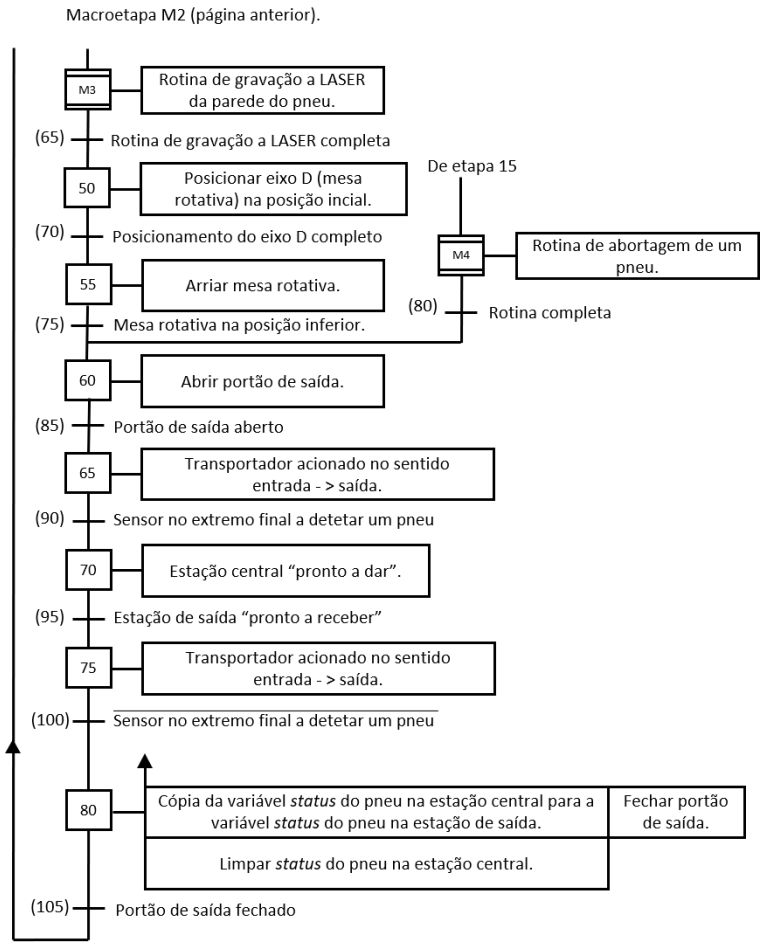
Caso este erro se dê, deve-se começar por limpar o erro (através da botoneira *acknowledge*) e a máquina apenas pode proceder ao pneu seguinte caso o operador confirme a abortagem através da HMI. Esta confirmação faz com que o *grafcet* 4 seja forçado à etapa E4 (etapa inicial da macroetapa M4 - rotina de abortagem[16]).

[16]Esta rotina será descrita no final desta secção.

Este comando está descrito no *grafcet* 0: mesmo que o *grafcet* 4 esteja congelado numa etapa, ao ser retomado, retomará na etapa E4.



(a) Parte 1 de 2.



(b) Parte 2 de 2.

Figura 4.15: *Grafcet* 4 - Ciclo automático da estação central.

Se as dimensões do pneu estiverem dentro do intervalo definido (condição que faz com a variável *DimensõesOK* tome o valor *True*), então o portão de entrada pode ser fechado e procede-se à energização do transportador. Quando o codificador do transportador chegar à posição em que o pneu deve ser centrado para se proceder à gravação, o transportador é desenergizado. Aqui assume-se que não haverá escorregamento maior do pneu que o leve a uma falha na operação de centragem que se segue.

O eixo A (responsável pela centragem) é energizado no sentido de arrastar o pneu até ao centro do transportador, utilizando um modo de controlo em binário, com velocidade limitada. Há duas razões para se ter adoptado esta estratégia:

- Se fosse utilizado controlo em posição, e o pneu fosse de uma dimensão externa inesperada, estar-se-ia a fazer uma força elevada no pneu ao comandar o eixo para uma dada posição;

- Se fosse utilizado somente controlo de binário (sem limite de velocidade), poder-se-iam alcançar valores de velocidade inesperados criando um choque com o pneu.

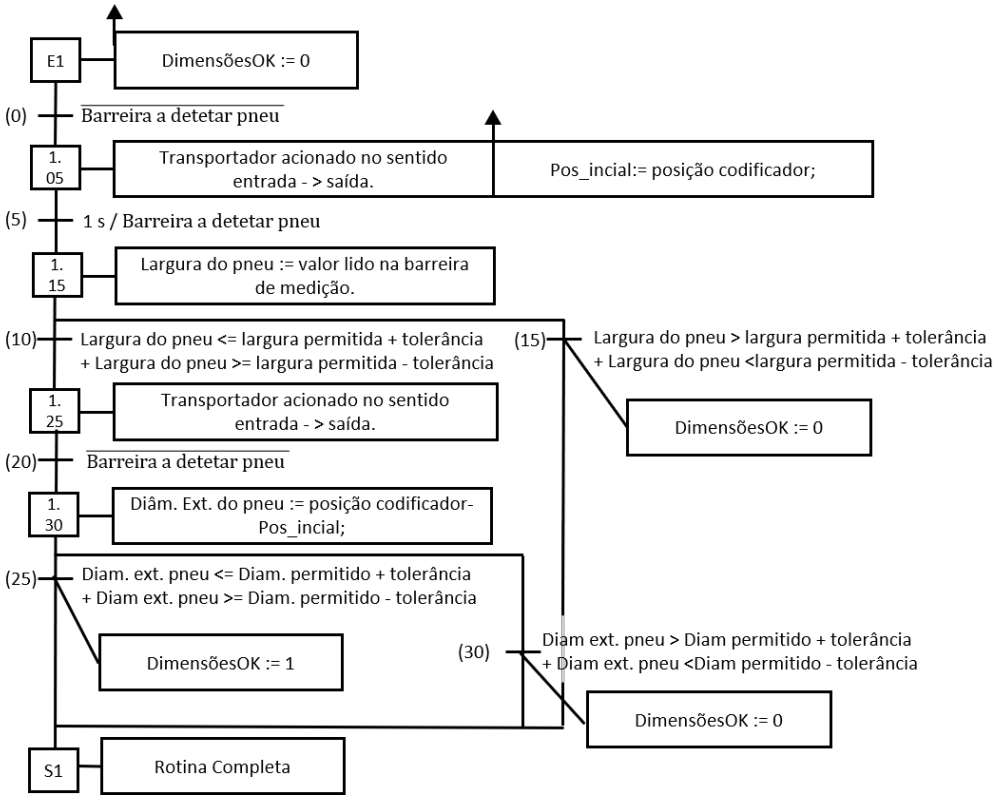


Figura 4.16: Expansão da macroetapa M1 - Rotina de verificação da largura e diâmetro externo do pneu.

A solução utilizada é uma funcionalidade já implementada pela própria RA e disponível de fábrica através da configuração dos respetivos parâmetros no *drive* do servomotor, chamada de *Speed-Limited Adjustable Torque* (SLAT).

Quando o eixo A alcança uma dada posição, assume-se que o pneu já está devidamente centrado e pode-se proceder ao recuo do mesmo (etapa 40). Seguidamente, a mesa é erguida e procede-se à rotina de deteção da referência do pneu utilizando a câmara, representada pela macroetapa M2 (figura 4.17).

A macroetapa M2 começa com o cálculo das posições que devem ser tomadas pela câmara nos eixos B (vertical) e C (horizontal), cujas fórmulas já foram determinadas na subsecção 4.2.4 - Interface com a câmara. De seguida, posiciona-se a câmara na posição de trabalho, e o comando de início da rotina da própria câmara é dado assim que esta esteja pronta a iniciá-la. No mesmo instante, a mesa é posta em rotação através do acionamento do eixo D.

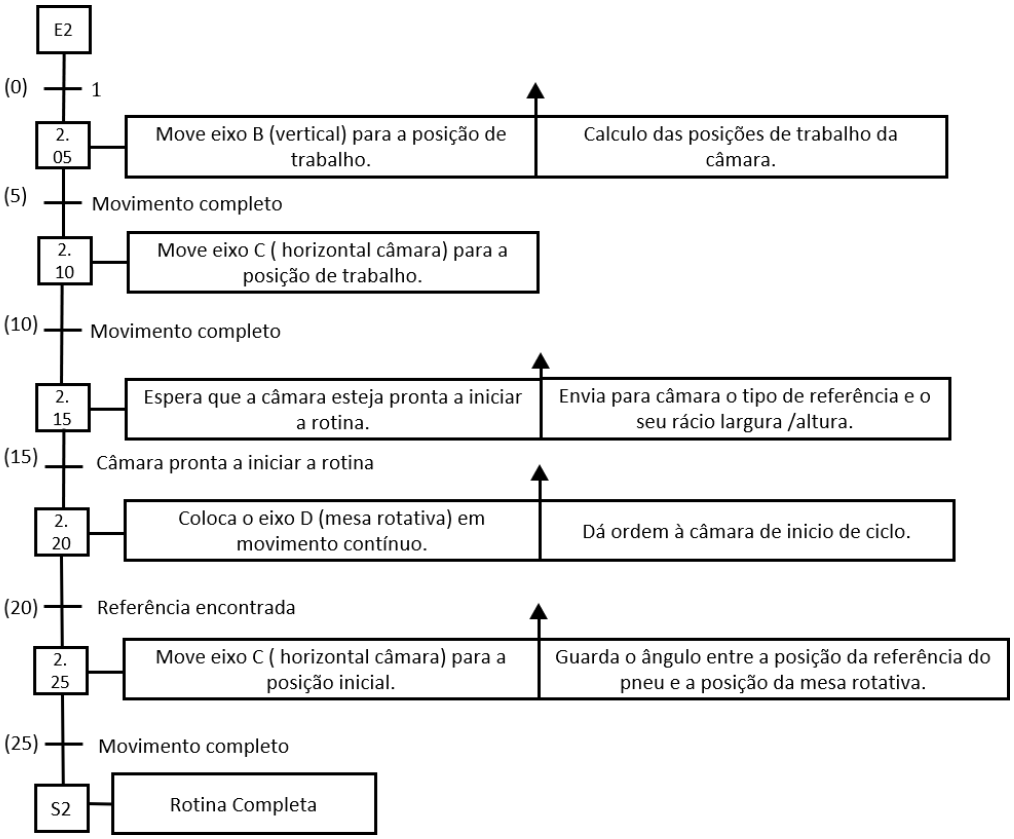


Figura 4.17: Expansão da macroetapa M2 - Rotina para determinação da posição da referência do pneu.

Após a referência ter sido encontrada, procede-se ao retorno do eixo C (horizontal da câmara) à posição inicial e a rotina considera-se completa, permitindo avançar para a etapa seguinte - macroetapa M3. A câmara devolve o ângulo entre a posição da referência do pneu e a posição do codificador montado no eixo D, pelo que este valor é guardado e será utilizado para posicionar a mesa rotativa durante a gravação.

Segue-se a macroetapa M3 (figura 4.18), que corresponde à rotina de gravação a LASER. Tal como a anterior, a rotina começa com o cálculo das posições de trabalho dos componentes, neste caso os eixos B (vertical), C (horizontal LASER) e D (rotação da mesa). A rotina inicia-se com a movimentação ordenada destes eixos para as respetivas posições de trabalho. A maneira como funciona o LASER está descrita na subsecção 4.2.2 - Interface com o LASER.

É dada uma ordem ao LASER para entrar em modo de gravação, e espera-se que ele fique pronto para a gravar. Nessa altura, é dado o comando de início através de uma saída digital do PLC e fica-se à espera de que a gravação seja terminada. Seguidamente, dá-se ordem ao LASER para sair do modo de gravação e retorna-se os eixos C (horizontal LASER) e B (vertical) para as posições iniciais, completando a macroetapa M3.

Voltando ao *grafcet* 4, reposiciona-se o eixo D na posição inicial e arria-se a mesa rotativa. O portão de saída é aberto e é feita a transferência do pneu para a estação seguinte, tal como foi feito da estação de entrada para a estação central. Ao deixar de sentir o pneu nesta estação, o seu *status* é copiado para a estação seguinte e o

portão é fechado, retornando à etapa inicial.

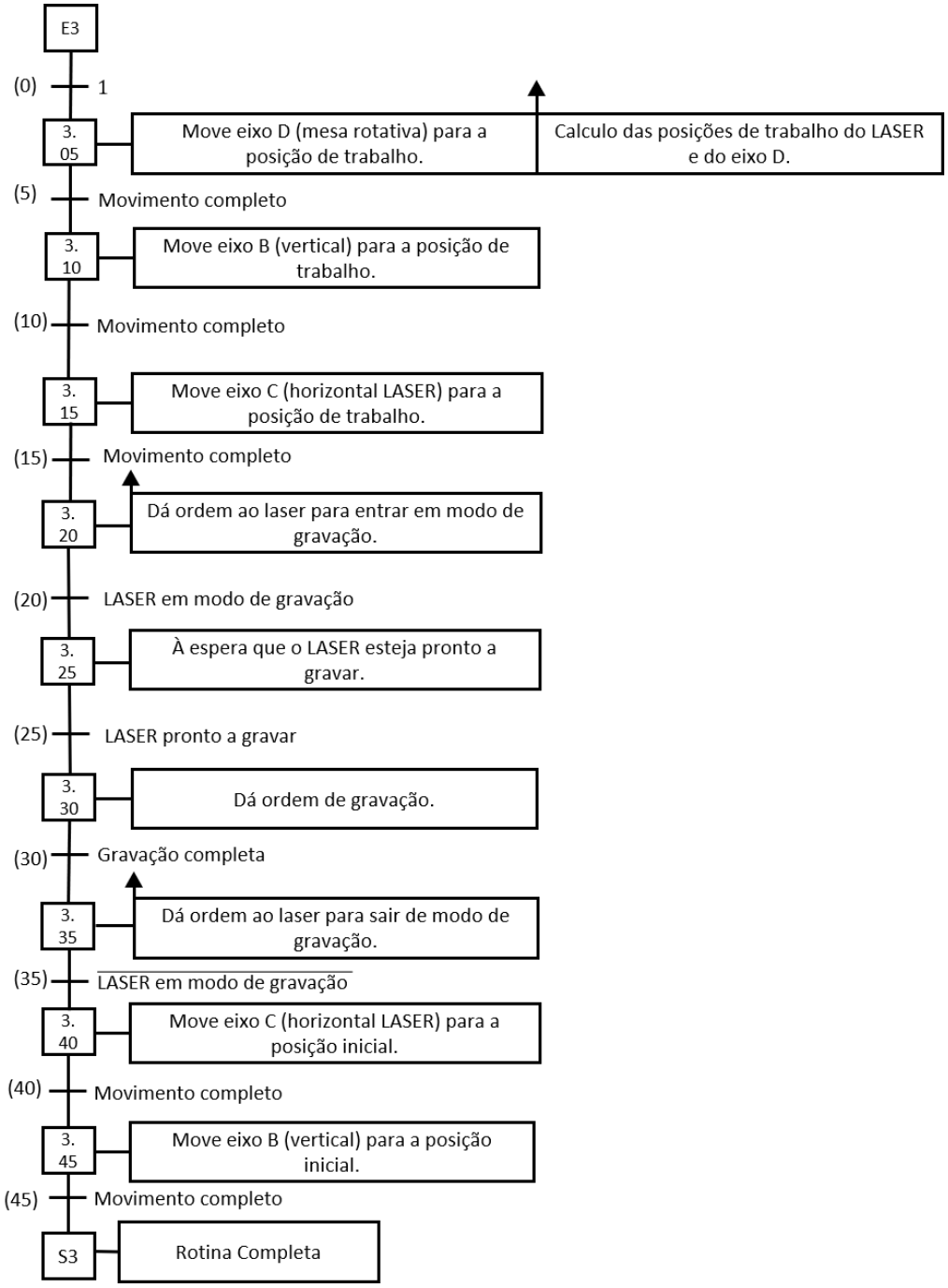


Figura 4.18: Expansão da macroetapa M3 - Rotina de gravação a LASER na parede do pneu.

Resta apenas descrever a rotina de abortagem de um pneu, que corresponde à macroetapa M4 e cuja expansão está representada na figura 4.19. Esta rotina começa por dar o *status* de abortado ao pneu em questão e procede à colocação de todos os eixos numa posição inicial, onde é garantido que não haverá colisões com o pneu (a mesa apenas é rodada para a sua posição inicial se já não estiver na posição

inferior). Finda esta rotina, dá-se a transição 80 do *grafcet* 4 e o pneu é expelido da estação central.

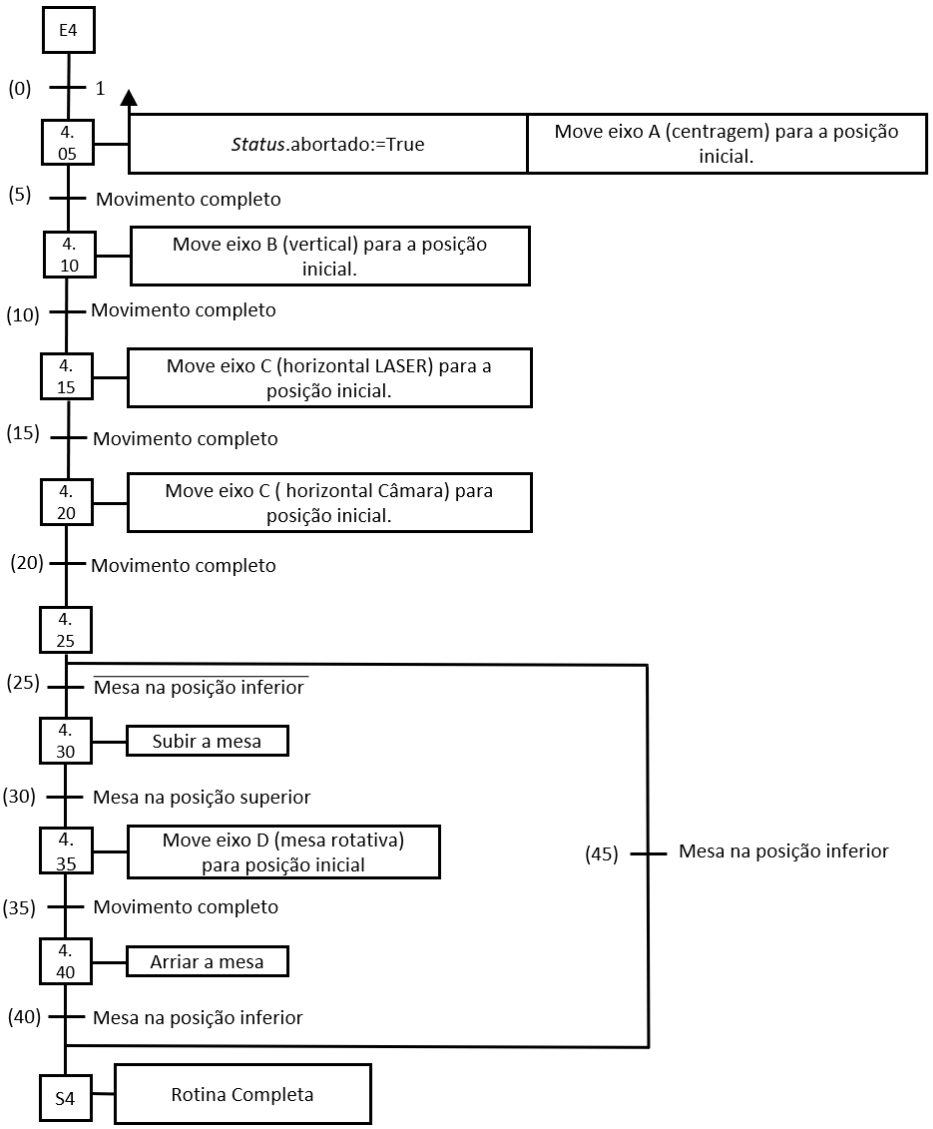


Figura 4.19: Expansão da macroetapa M4 - Rotina de abortagem de um pneu.

Termina assim a apresentação do comportamento programado da máquina.

4.4 Aplicação prática e testes

Esta secção tem como objetivo expor a aplicação do ponto de vista prático dos conceitos descritos nas secções anteriores. Apesar de a máquina não ter sido fisicamente construída, todo o código foi testado dentro do possível. Para tal, recorreu-se aos seguintes componentes:

- Um PLC da RA, com capacidade de comando de movimento - Compactlogix 5370;

- Um *drive* de servomotor da série Kinetix 5500 com fonte de alimentação e módulo juntos num só componente;
- Um servomotor da série VPL da RA;
- Um módulo de controlo de rolos e um rolo da marca Interrol;
- O *software* de simulação de ambiente industrial factoryIO;
- Uma base de dados criada numa instância do Microsoft SQL server, com apenas duas tabelas (entrada e saída);
- Um simulador para o LASER criado em C# pelo autor;
- Um servidor e cliente FactoryTalk SE a correr no mesmo computador.

Apesar de todos estes componentes não serem suficientes para emular uma máquina na totalidade, a empresa reconheceu que isto seria suficiente para validar o trabalho realizado, visto que a máquina acabou por não ser construída a tempo.

As subsecções seguintes dizem respeito a aspetos da implementação do código no PLC, programação de erros do PLC e por fim serão descritas as soluções utilizadas para simular o processo.

4.4.1 Organização de programas e rotinas do controlador

Visto que se está perante três estações fisicamente diferentes, com vários componentes semelhantes (i.e: sensores de entrada/saída e transportador), que teriam de interagir umas com as outras, optou-se por colocar cada uma num programa diferente. Relembra-se que cada programa é composto por várias rotinas e tem as suas próprias variáveis e parâmetros.

Como sugerido pela empresa, cada estação deve estar preparada para funcionar como se fosse uma estação independente. Desta forma, os programas tornam-se inerentemente ligados à estação física, permitindo alterar a disposição da máquina de forma modular. Por exemplo, se fossem colocadas duas estações centrais em série, bastava copiar e colar dois programas idênticos, e referenciar devidamente os seus parâmetros de entrada e/ou saída.

Assim, pode-se tratar cada estação como um objeto com os devidos parâmetros de entrada e/ou saída para o controlar. Note-se que estas considerações têm grande capacidade de extrapolação para programação de outras máquinas, visto que uma máquina pode ser dividida em várias estações funcionais com interfaces entre elas.

Apesar de cada estação se conseguir gerir independentemente, é necessário um programa que faça de maestro para as orquestrar. Este será denominado o *MainProgram*[17], e tem como funções:

- Constituir um controlo central que dá ordens às estações da máquina, em função do status de cada estação independente. Por exemplo: ordem de início, ordem de suspensão, ordem de reinício;

[17] O comportamento descrito no *grafcet* 0 está programado no *MainProgram*.

- Realizar a rotina de reinicialização da máquina que funciona como uma única estação durante esta rotina;
- Gerir entradas/saídas trocadas com o PLC que controla os transportadores da fábrica;
- Gerir erros gerais da máquina;
- Gerir certos procedimentos que se dão no SCADA.

Os programas estão todos incluídos na mesma tarefa contínua (chamada *MainTask*), porque se entendeu que têm todos a mesma prioridade. Na figura 4.20 pode-se ver de que maneira isto é materializado no Studio 5000: estão presentes as três estações e o *MainProgram* que faz de maestro. Na mesma figura, a estação central está apresentada como *Work_Station* porque é onde decorre o processo de gravação.

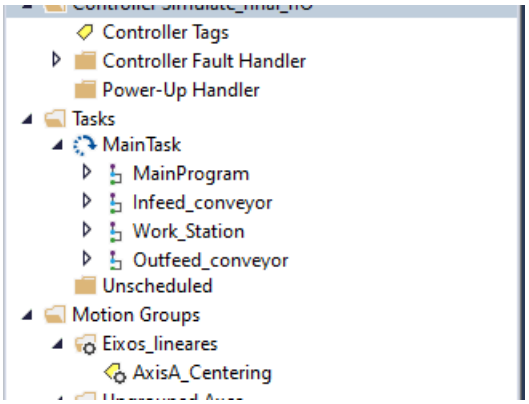


Figura 4.20: *Printscreen* do Studio 5000 onde se vê a organização de programas na *MainTask*.

Cada programa contém subrotinas, que são unidades de programação de menor nível, onde é literalmente implementado o código. Durante a programação da máquina, recorreu-se às linguagens Ladder, *Sequential Function Chart* (SFC) e *Structured Text* (ST) dependo da aplicação. O Ladder foi usado principalmente em subrotinas e o ST em AoIs; já o SFC foi utilizado somente nos ciclos de processo das estações em modo automático.

A organização de cada programa é um assunto muito importante, que deve seguir uma regra para que um futuro editor possa consultá-lo intuitivamente. Assim, e à luz de um exemplo dado pela empresa, organizou-se cada programa de acordo com a figura 4.21.

O nome de cada rotina dispõe de uma letra antes para que estas fiquem organizadas por uma ordem lógica. Passa-se a descrever a função de cada uma:

- A_Organization: Fazer a chamada das restantes rotinas pela ordem que são apresentadas nesta lista;
- B_Configuration: Inserção parâmetros da máquina, por exemplo velocidade de *jog* dos servomotores, velocidade de comando dos transportadores, tempo de timeout, tolerâncias dimensionais dos pneus, entre outros;

- C_Operation_Status: Inserção da lógica de comando da estação, descrita no *grafcet* 1 ao nível de cada estação;
- D_General_Errors: Configuração e gestão de erros da estação;
- E_Input_Buffer: Caso seja necessário fazer passagens por valor de entradas;
- F_Interlock: Programação da lógica de *interlock*;
- G_Automode: Ciclo da estação em modo automático;
- H_Commands_AoIs: Inserção de AoIs (por exemplo de comando de servomotores e cilindros pneumáticos);
- I_Communication: Comunicação com equipamentos, por exemplo o LASER;
- J_HMI_Control: Lógica específica de controlo de algumas situações do SCADA a correr na HMI;
- K_Output_Buffer: Caso seja necessário fazer passagens por valor de saídas.

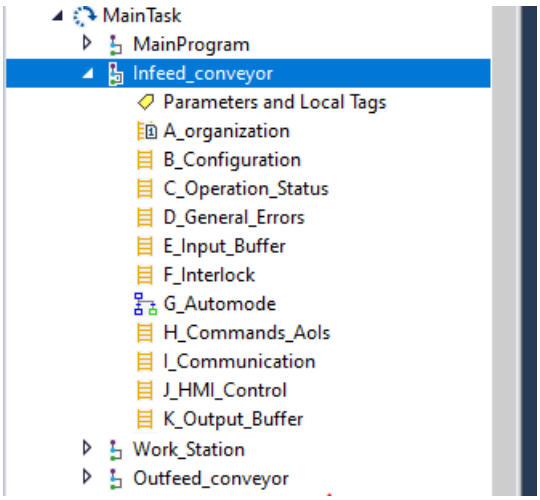


Figura 4.21: *Printscreen* do Studio 5000 onde se vê a organização de subrotinas em cada programa.

Adicionalmente, podiam ser adicionadas outras rotinas que se entendessem necessárias, conforme a aplicação.

4.4.2 Programação de suporte a situações de erro

Erros são normalmente resultado de falhas de componentes ou situações imprevistas. Como o PLC normalmente não consegue resolver este tipo de situações autonomamente, deve-se prever um dado comportamento em caso de erro. Como é evidente na secção 4.3 - Descrição comportamental do funcionamento da máquina, o ciclo é suspenso em caso de erro e fica à espera de que o operador o reconheça.

Caso o erro seja reconhecido e se resolva, a máquina pode retomar o seu ciclo. A situação pode ser um simples cabo mal ligado ou uma falha momentânea do um certo componente.

Os erros podem surgir de equipamentos ou do próprio ciclo. Os erros de ciclo podem aparecer devido a eventos ou resultados inesperados, ou então por *timeout*. O *timeout* significa que o pneu está demasiado tempo numa dada etapa do ciclo, que leva à suspeita de que alguma coisa de errado aconteceu. O tempo de *timeout* foi programado para ser ajustável aquando da fase de testes da máquina. Segue-se uma discretização das várias situações de erro programadas:

- Erro na receita recebida da tabela de entrada;
- Erro de comunicação com o servidor SQL;
- Erro na rotina de reinicialização;
- Erro dos transportadores;
- Erro de *timeout* dos ciclos em modo automático;
- Erro na leitura das dimensões do pneu;
- Erro de movimento inesperado do pneu nos ciclos automáticos;
- Erro da câmara;
- Erro do LASER;
- Erro dos cilindros pneumático;
- Erro dos servomotores.

Deste modo, consegue-se alertar o operador para o componente individual no qual o erro se deu, e qual a sua causa. Isto é feito através de um alarme que aparece na HMI, que será descrito na secção 4.5 - Criação dos ecrãs do SCADA

4.4.3 Soluções utilizadas para simular o processo

Factory IO

O Factory IO é um *software* para criação de ambientes industriais que permite criar cenários com imensos dispositivos existentes na indústria. Neste caso, recorreu-se a transportadores, botoneiras, cilindros pneumáticos e sensores retrorrefletivos para poder simular toda a lógica sequencial do processo.

O ambiente do Factory IO utilizado pode-se observar na figura 4.22. Nesta figura, são visíveis três transportadores, cada um representante de uma estação. Em vez de pneus, recorreu-se a caixotes que são igualmente detetáveis por sensores e mensuráveis na barreira de medição.

Os objetos paralelipipédicos brancos e cinza são chamados de *pushers* e representam cilindros pneumáticos sensorizados nos dois extremos, com controlo bidirecional. Os três *pushers* visíveis na figura referida correspondem aos portões das estações de

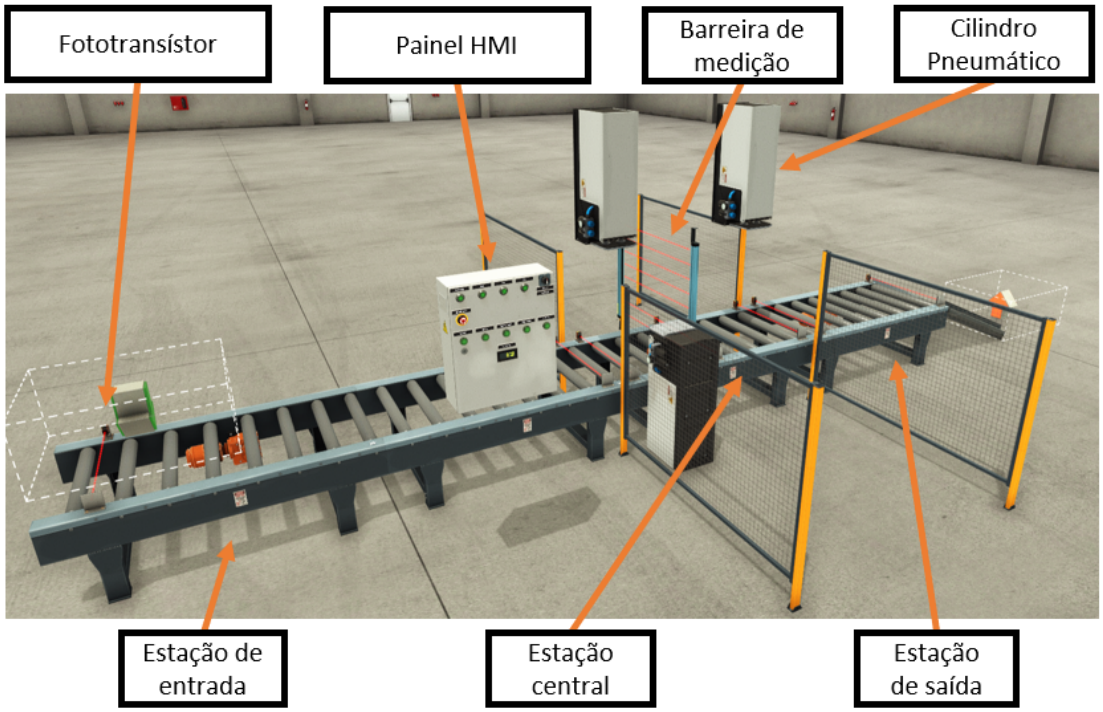


Figura 4.22: Cenário criado no Factory IO para simulação do processo.

entrada/saída e ao cilindro para elevação da mesa rotativa. Os componentes que não foram possíveis de incluir nesse cenário são o LASER, a câmara e os eixos servomotorizados.

Simulador LASER

Como estava disponível em manual o protocolo de comunicação do LASER com o PLC, o autor viu uma oportunidade de criar um simulador para o LASER. A sua finalidade seria permitir testar a AoI criada para o LASER, sem ser com um simples terminal TCP/IP. O problema do terminal é que torna o processo de teste lento e confuso.

Optou-se por fazer este simulador em C#, no Visual Studio, no qual já se conhecia bem as capacidades de comunicação por *sockets* TCP/IP. Para não sobrecarregar o computador com janelas de formulários, decidiu-se que apenas deveria aparecer a janela do terminal de comandos (figura 4.23).

O objetivo deste simulador é receber os pedidos do PLC e responder como se fosse o próprio LASER. Por exemplo, se o PLC pede para entrar em modo de impressão com o ficheiro "ABCD.xml", o LASER responderá com confirmação de que entrou no modo de impressão. Quando o PLC lhe fizer um pedido do *STATUS*, o LASER deverá dizer que está em modo de impressão com o ficheiro "ABCD.xml".

A lógica seguida é que o LASER deverá dar uma resposta em função daquilo que lhe é pedido e do seu estado atual. Caso a conexão com o PLC seja perdida ou ainda não tenha sido estabelecida, o LASER estará simplesmente à espera de uma conexão.

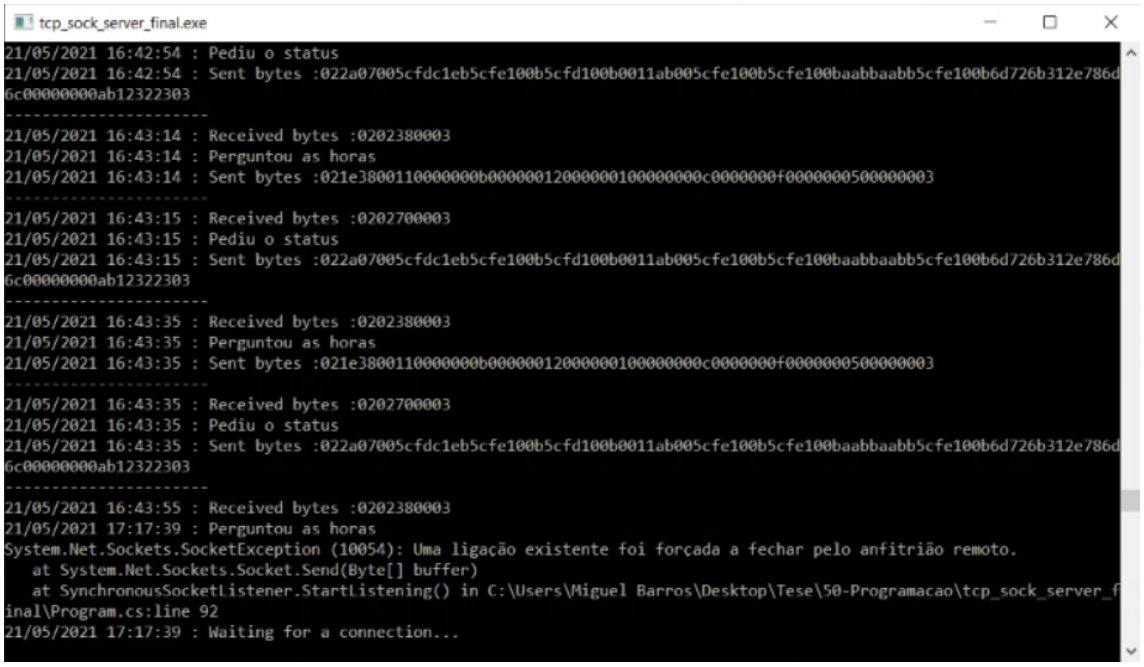


Figura 4.23: *Printscreen* do terminal do simulador do LASER.

O comportamento da própria ação de gravação não pode ser simulado, pelo que isto não foi coberto. Como já foi explicado na secção 4.2 - Considerações sobre os componentes da máquina, quando o LASER termina o seu processo, colocará a *True* o valor de uma saída digital específica, o que seria impossível para este programa em C#[18].

À medida que o PLC vai fazendo pedidos e o LASER vai respondendo, é escrito o resultado destas operações no terminal do C#. Isto permite ao programador acompanhar a informação que é trocada à medida que o processo decorre.

Na figura 4.23, pode-se ver que o PLC estava a fazer um pedido da hora interna do LASER e do seu *status* a cada 20 s. Como a comunicação entre o PLC e o LASER foi interrompida entretanto, essa informação também foi devidamente anunciada no terminal.

4.5 Criação dos ecrãs do SCADA

Finda a descrição da solução implementada de programação para controlo da máquina, proceder-se-á à apresentação da solução criada para os ecrãs do SCADA, cuja visualização decorre no monitor da HMI. Na prática, a criação dos ecrãs também foi feita após a a programação da lógica. A criação de uma visualização constitui um desafio bastante diferente da programação do PLC no sentido que se trata da interface diretamente com o utilizador final.

Uma interface visual consiste num conjunto de ecrãs (ou *displays*) que estão abertos ao mesmo tempo sobre uma tela. Em cada ecrã existem elementos de entrada, de saída e de cenário. A navegação entre ecrãs é feita através de botões programados.

[18]Para testar, fez-se com que a etapa de gravação terminasse ao fim de 3 s.

A tela é um espaço em branco onde são abertos ou fechados os vários ecrãs. Os ecrãs podem ser essencialmente de dois tipos: do tipo que se sobrepõe aos outros, ou do tipo que provoca o fecho dos outros que estariam a ocupar o seu espaço. A conjugação de ambos os tipos de ecrã depende da estratégia adotada e não existe propriamente uma regra única.

Os elementos de entrada são aqueles que servem para receber informação do utilizador, já os de saída servem para informar o utilizador. Os de cenário são geralmente elementos estéticos ou informativos que contribuem para a boa apresentação de cada ecrã.

Decidiu-se que os ecrãs devem ser divididos por funcionalidade. Este tipo de divisão permite ao utilizador ter imediatamente um palpite de onde encontrar a informação que procura sem ter que memorizar nada. Dentro dos ecrãs, devem ser feitos grupos de informação em vez de a ter espalhada.

Para apresentar informação que se repete várias vezes, são utilizados os chamados *global objects* e constituem o equivalente gráfico aos objetos da programação normal: têm parâmetros de entrada/saída, variáveis internas e podem correr as suas próprias funções. A sua utilidade prática torna-se evidente com o exemplo abaixo.

Imagine-se que se quer criar uma pequena área no ecrã para controlo de um servomotor. Devem ser colocados objetos gráficos para apresentar a velocidade de rotação, a sua posição e ainda alguns botões para comandar o servomotor, agrupados num painel. As variáveis controladas por estes objetos devem ser devidamente ligadas às respetivas do PLC que dizem respeito ao servomotor. Caso existam dois servomotores, então seria necessário copiar e colar o painel anterior e reprogramar cada uma das variáveis do objeto.

Ao criar um *global object*, pode-se alocar às variáveis de velocidade, posição e botões um *placeholder* em vez das variáveis de um determinado servomotor. Depois, ao criar cada instância do *global object*, basta escolher qual a variável (ou conjunto de variáveis) do servomotor que precisam de ser ligadas, em vez de reprogramar cada um dos objetos. Caso o programador quisesse fazer uma alteração a ambos os painéis ao mesmo tempo, bastava editar o *global object* base. Desta forma, o tempo necessário para a programação é reduzido exponencialmente por cada instância que se utilize.

Nas subsecções que se seguem, serão descritos os vários ecrãs criados para esta aplicação. A leitura deve ser acompanhada da visualização dos ecrãs, representados em figuras ao longo do texto e disponíveis em dimensões maiores no anexo C. Porém, antes de passar à análise individual, atente-se à forma como é feita a navegação entre os vários ecrãs.

Para navegar entre ecrãs, optou-se por um conceito semelhante à barra de tarefas do próprio Windows (disponível em todas as páginas do anexo referido). O nome dos ecrãs disponíveis aparece em baixo, permitindo ao utilizador navegar entre eles através de um *click*. Aquele que estiver aberto é cromaticamente destacado dos restantes.

Na barra de navegação foram adicionados ícones com significado global para cada um dos ecrãs. Desta forma, os ecrãs podem ser identificados pelo seu símbolo ou pelo seu nome. No canto inferior direito existe um pequeno espaço para colocar símbolos de notificações. Até à altura, apenas foi implementado um símbolo de um

botão de emergência que pisca a 2 hz quando a segurança cai, e fica invisível quando esta é reposta.

Foram previstos cinco ecrãs principais, navegáveis entre eles:

- O ecrã *Main*:

  Contém informações pertinentes do processo e do estado das estações;

- O ecrã *Machine Components*:

  Onde se pode consultar os componentes da máquina, verificar o seu *status* individual e controlá-los em modo manual;

- O ecrã *Settings*:

  Onde se podem consultar e alterar alguns parâmetros da máquina;

- O ecrã *Inputs & Outputs*:

  Prevê-se que nesse ecrã sejam colocados os valores lidos nos módulos de entradas e saídas do PLC. Este ecrã não foi criado porque não fazia sentido criá-lo sem conhecer os componentes físicos utilizados;

- O ecrã *Alarms & Events*:

  Onde o operador pode consultar em detalhe informação sobre alarmes e eventos.

Dentro de cada ecrã principal, podem existir um ou mais ecrãs secundários que serão devidamente referidos nos próximos parágrafos. Existe apenas um ecrã que não está susceptível à navegação, o ecrã com o painel de cabeçalho. A navegação entre os ecrãs referidos é feita segundo a figura 4.24.

Na figura, as setas pretas ao centro indicam que a navegação entre os ecrãs principais não é sequencial, ou seja, o ecrã aberto é sempre aquele que se selecionar na barra de navegação em baixo. As restantes setas apontam para o ecrã que abre quando se carrega em determinados botões do ecrã de onde estas partem. No ecrã *Settings* existe uma barra lateral esquerda para navegar entre ecrãs secundários que também dizem respeito às definições.

Adicionalmente, o utilizador pode consultar informações sobre alguns componentes da máquina (por exemplo, de um servomotor, de um cilindro pneumático ou o LASER) através de um ecrã que se sobrepõe aos restantes e situa-se do lado direito da tela. Estes ecrãs contêm informação sobre o estado dos componentes em questão e é através dele que se pode fazer seu o controlo em modo manual. O acesso a estes ecrãs será explicado na secção relativa a cada ecrã principal.

Estes ecrãs podem ser consultados nas páginas 153, 154, 156, 158, 159, 160, 161, 165 e 166 do anexo C para todos os dispositivos que foram criados e correspondem a:

- Fluxo do processo das estações de entrada ou saída;

- Fluxo do processo da estação central;

- Painel de controlo de um transportador;
- Painel de monitorização da câmara;
- Painel de controlo de um cilindro pneumático;
- Painel de monitorização do LASER;
- Painel de controlo de um servomotor;
- Painel de monitorização do PLC da máquina;
- Painel de controlo do servidor SQL.

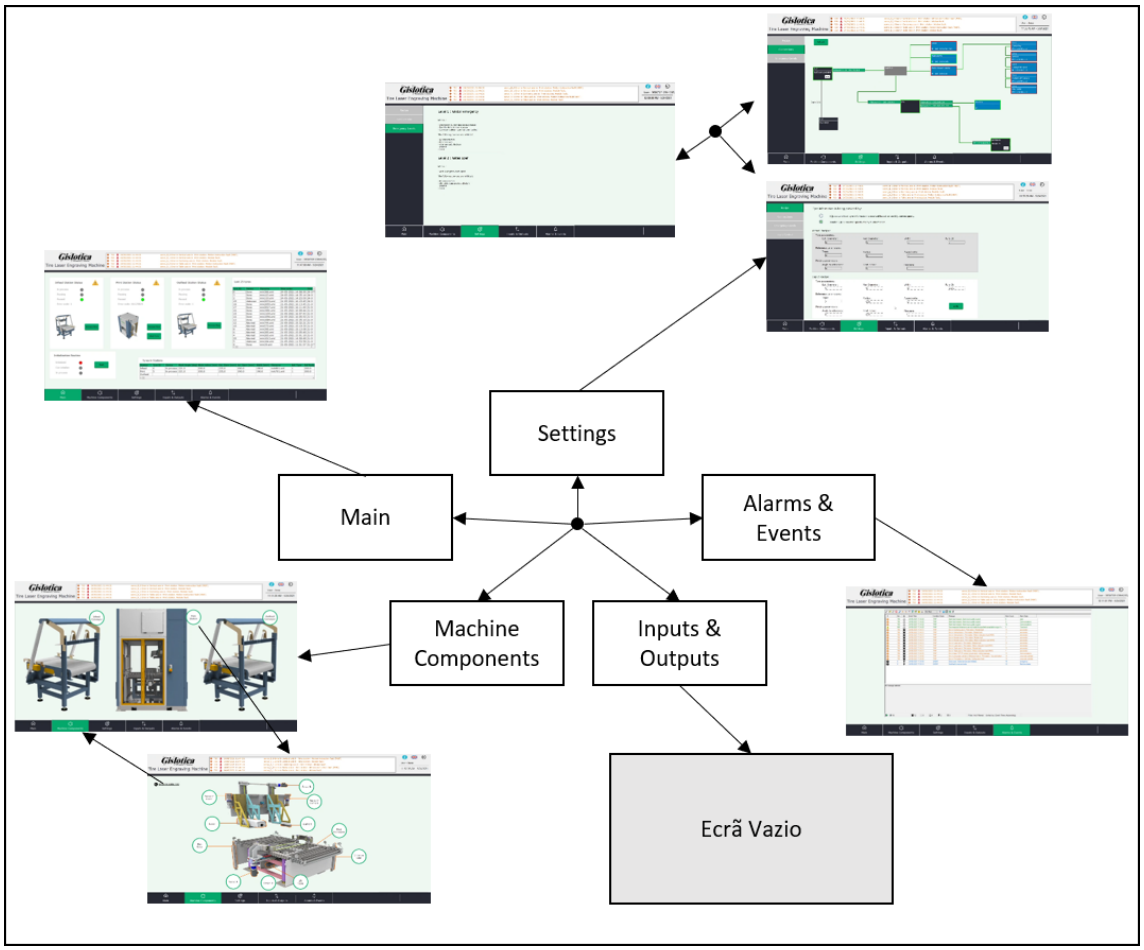


Figura 4.24: Esquema de navegação entre os ecrãs principais no FactoryTalk.

Para aqueles componentes da máquina que se repetem[19], foram criados *global objects* para facilitar a edição dos respetivos ecrãs de controlo e monitorização.

[19]Os componentes que se repetem são as estações, os cilindros pneumáticos, os transportadores e os servomotores.

4.5.1 Painel de cabeçalho

O primeiro componente da visualização a apresentar é o seu cabeçalho, que está sempre visível e aparece na zona superior do monitor. Na figura 4.25 é possível visualizar os seguintes elementos da esquerda para a direita:

- O logótipo da empresa:

  Serve para informar o utilizador de quem é o produtor da máquina;

- O nome da máquina:

  Serve para descrever em poucas palavras qual a finalidade desta máquina - gravação a LASER de pneus;

- Uma pequena área dedicada a alarmes:

  Para o utilizador ter sempre uma noção de qual é o erro mais pertinente no momento;

- A hora e a data:

  Um elemento bastante útil quando o cliente do FactoryTalk está a correr em ecrã completo (*fullscreen*);

- O nome do utilizador:

  Para o utilizador saber se está realmente a utilizar a sua conta;

- Um botão para obter informações sobre a empresa: Este botão serve para incluir informação genérica e contactos em caso de avaria (disponível na página 169);

- Uma bandeira para alterar a linguagem:

  Neste caso apenas foi implementado o inglês, mas ao carregar neste botão poderia alternar entre inglês e português, por exemplo;

- Um botão para fazer *log out*:

  Este botão permite ir para o ecrã de autenticação de utilizador.

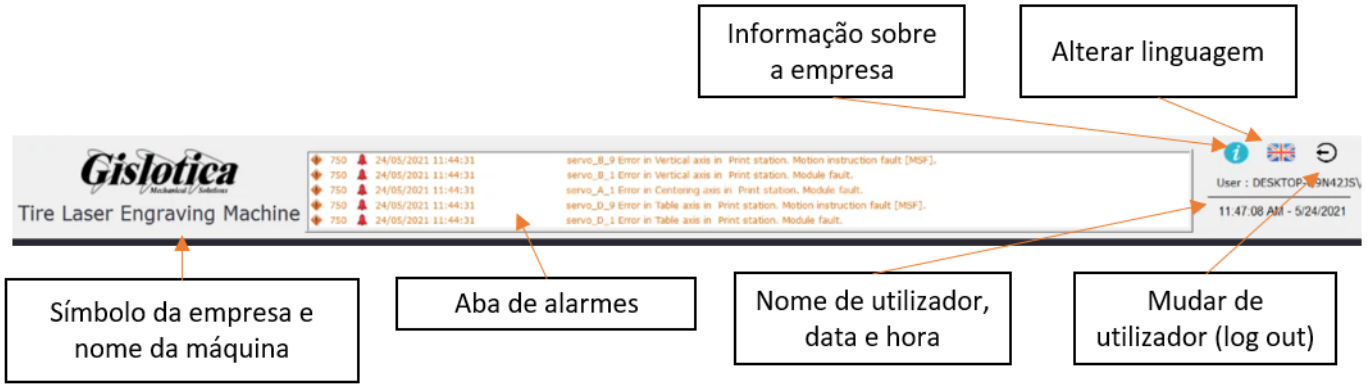


Figura 4.25: Painel de cabeçalho.

4.5.2 Ecrã *Main*

O primeiro ecrã a apresentar é chamado de *Main* (ou ecrã principal) e tem como objetivo apresentar informação pertinente ao leitor sobre o processo e sobre o estado da máquina: está representado na figura 4.26 e está disponível na página 152 do anexo C. Podem-se observar seis grupos de informação com um propósito distinto.

A informação sobre o estado da máquina é dada pelos três painéis de estação, cada uma com uma pequena imagem e nome da respetiva estação[20], permitindo identificar se a estação está em processo, suspensa ou a suspender e o código de erro atual. Adicionalmente, estes objetos contêm um botão *Process Flow* e outro para abortar o pneu (figura 4.27).

O fluxo do processo é um diagrama que mostra em que etapa da rotina de ciclo do modo automático a máquina se encontra, uma informação que é bastante útil para diagnóstico. Este diagrama pode ser observado no ecrã lateral que é aberto ao pressionar o respetivo botão, e está disponível nas páginas 153 e 154 do anexo C. O ecrã do fluxo processo tem aspeto igual para as estações de entrada e saída, mas diferente para a central.

O painel abaixo à esquerda contém informações sobre a inicialização da máquina, que pode ser vista em detalhe na figura 4.28. Os indicadores luminosos permitem saber se é necessário correr a rotina de reinicialização, se esta rotina pode ser corrida e se a rotina está em processo.

Foi adotado um certo significado para as cores utilizadas nos sinalizadores luminosos. Este objectos são normalmente utilizados para representar o valor de variáveis booleanas, e podem ter duas cores (uma corresponde ao estado *True* e outra ao estado *False*).

A codificação implementada foi a seguinte: caso o sinalizador apresente uma cor verde, quer dizer que o estado representado pelo valor da variável tem uma conotação positiva. Caso a cor seja vermelha, o estado tem uma conotação negativa. Finalmente, caso a cor seja cinza, quer dizer que o estado tem uma conotação neutra,

[20] Utilizou-se a fotografia da obra da empresa com capacidade de viragem do pneu para representar as estações de entrada/saída porque não existia uma fotografia destas estações sem o mecanismo de viragem.

isto é, nem positiva nem negativa.

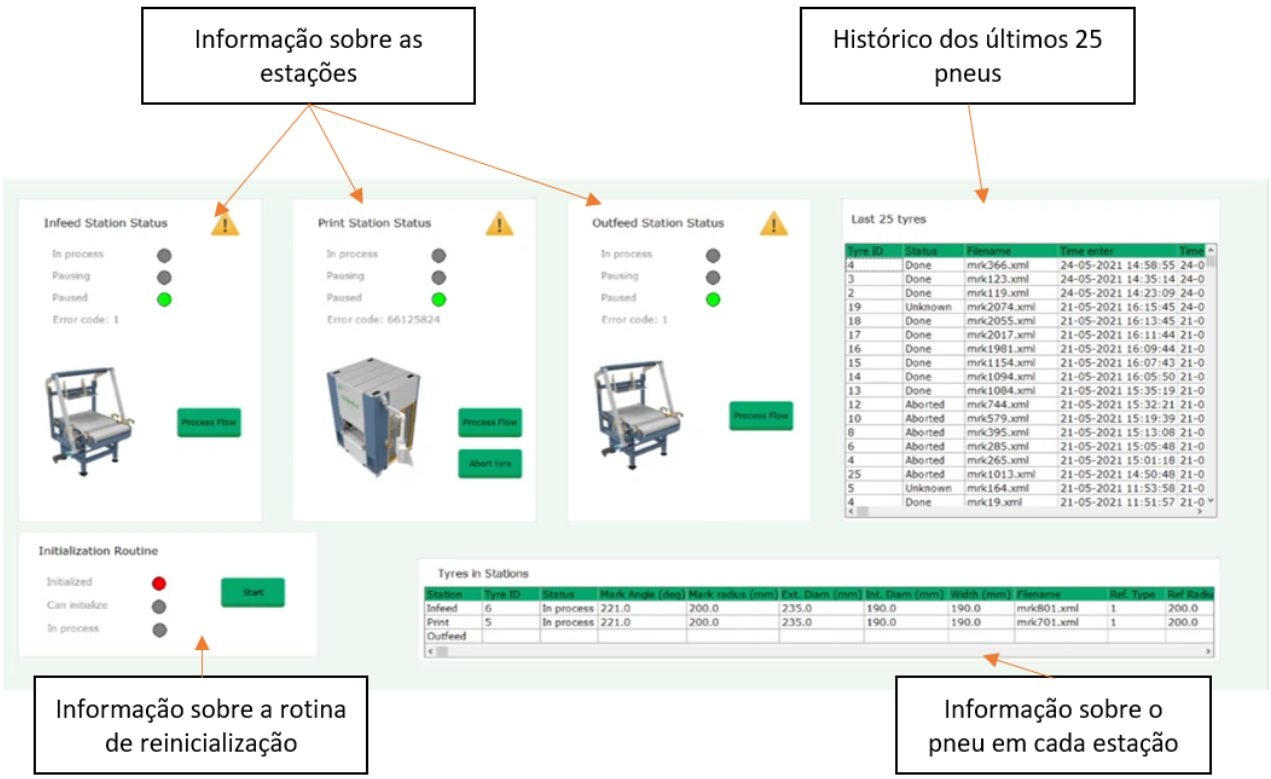


Figura 4.26: Ecrã *Main*.

Por exemplo, um sinalizador que indica se o ciclo de uma estação está suspenso toma o valor verde em caso afirmativo e cinza em caso negativo. Já o sinalizador que indica se a máquina está devidamente inicializada, toma o valor vermelho quando não está inicializada para que o operador entenda que isto é uma condição necessária para o funcionamento, e que deve ser corrigida. Quando a máquina está devidamente reinicializada, o mesmo sinalizador toma a cor verde para indicar que este é o estado positivo no sentido do decorrer do processo.

No caso da figura 4.28, pode-se verificar que a máquina necessita de fazer a rotina de reinicialização e que a rotina não pode ser corrida, neste caso, porque o seletor manual/automático se encontrava na posição manual. O botão *Start* serve para o operador dar ordem de início da rotina de reinicialização.

Voltando à figura 4.26, em cima à direita está uma tabela com os últimos vinte e cinco pneus que passaram na estação central. Esta tabela contém o ID dos pneus, o resultado da operação, o nome do ficheiro da imagem a gravar nos pneus, o tempo em que entraram e o tempo em que saíram da estação central. Através desta tabela, o operador pode ter uma noção do histórico do processo e do tempo demorado em cada pneu.

Em baixo, contém uma tabela com os pneus atualmente em cada estação. É apresentada toda a informação conhecida sobre cada pneu, permitindo ter uma noção do que está a acontecer sem ter de ver realmente o pneu. Como em ambas

as tabelas existe mais informação do que aquela que é constantemente pertinente, incluiu-se as barras para *scrollar* entre colunas e linhas.

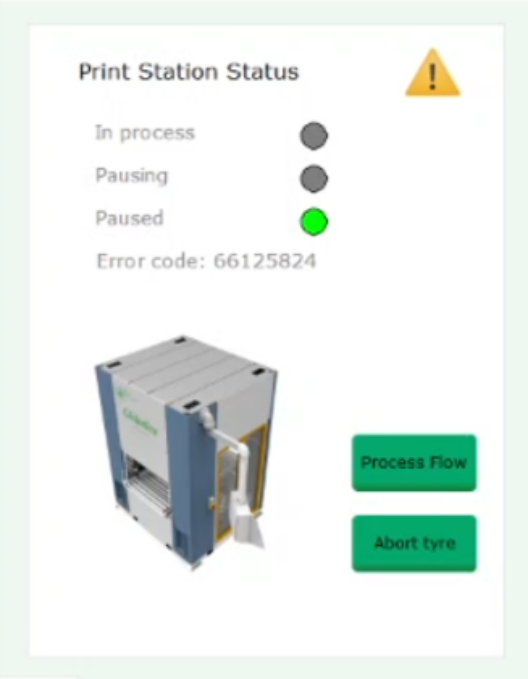


Figura 4.27: Painel com informação sobre a estação central.



Figura 4.28: Painel com informação sobre a rotina de reinicialização.

4.5.3 Ecrã *Machine Components*

Este ecrã permite ao operador abordar um a um os vários componentes existentes na máquina e está disponível na figura 4.29 e na página 155. A consulta de cada componente é conseguida através da navegação até à respetiva estação e *click* no respetivo balão. Na figura, destacam-se as três estações da máquina, cada uma com o respetivo balão devidamente legendado para *clickar*.

Como as estações de entrada e saída apenas têm um componente a controlar (o transportador de rolos), ao *clickar* no seu balão é aberto o ecrã de controlo dos

transportadores, disponível na página 156. Esse ecrã de controlo permite verificar o estado atual do transportador e utilizar os botões verdes para o comandar; adicionalmente tem um aviso a amarelo para o operador compreender que só pode comandar o transportador se o seletor manual/automático estiver na posição manual.

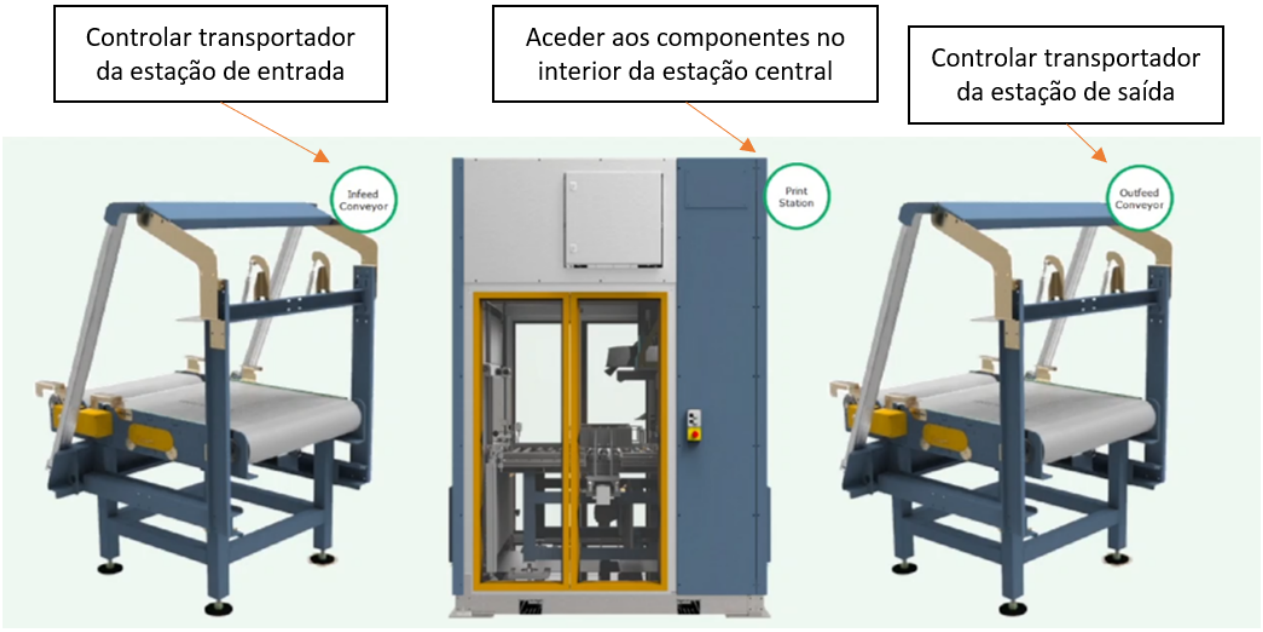


Figura 4.29: Ecrã *Machine Components*.

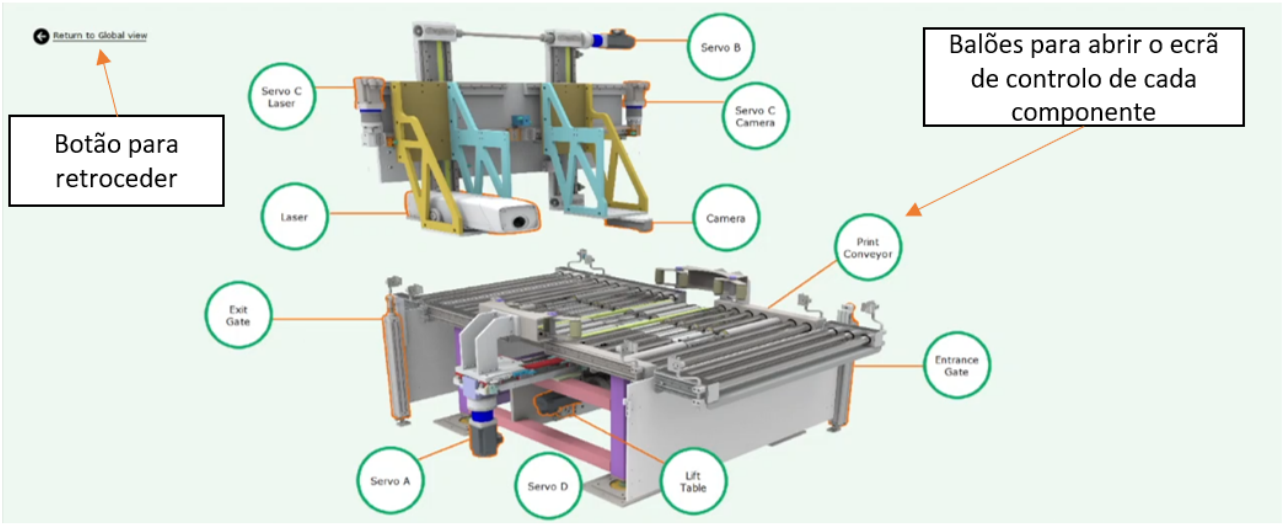


Figura 4.30: Ecrã *Machine Components*.

Pressionando alternativamente o balão "Print Station", é aberto o ecrã apresentado na figura 4.30 e na página 157. Este ecrã contém uma imagem do interior da máquina, com os vários componentes controláveis destacados a cor-de-laranja; foram colocados balões para consultar os ecrãs de controlo de cada um dos componentes, disponíveis nas páginas 158 a 161.

Não será feita aqui a descrição exaustiva dos objetos presentes em cada um dos ecrãs de controlo porque o leitor pode consultá-los mais facilmente de forma visual. No canto superior esquerdo deste ecrã foi colocado um botão de retroceder que permite voltar ao ecrã anterior, representado na figura 4.29.

4.5.4 Ecrã *Settings*

Clickando no respetivo botão da barra de navegação, é aberto o ecrã *settings* que está representado na figura 4.31 e na página 163. À esquerda, está disponível uma aba de navegação entre os vários tipos de definições disponíveis. Na aba receita (o primeiro ecrã aberto), existe um *radio button* que permite ao utilizador alternar entre receita vinda do servidor SQL ou fazer a introdução manual da informação acerca do pneu que entrará na estação de entrada.

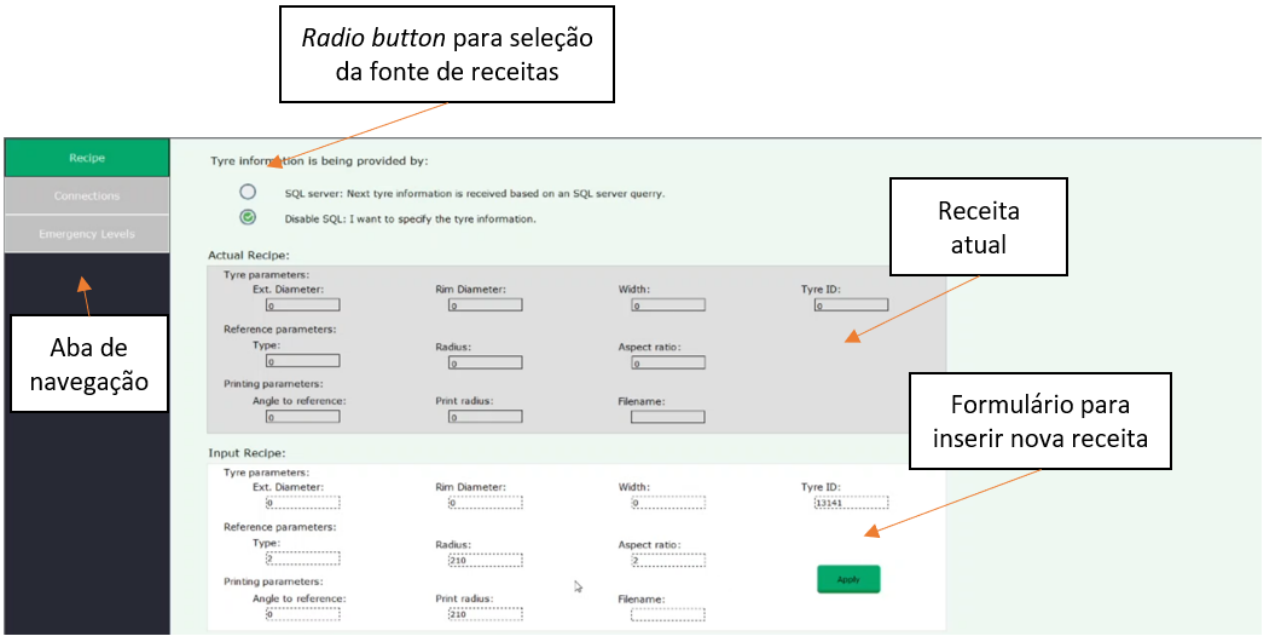


Figura 4.31: Ecrã *Seetings*, aba *Recipe*.

Apesar de não ter sido representado nos *grafcets*, foram implementados mecanismos de *bypass* das etapas em que é necessário interagir com o servidor SQL. A razão para isto ter sido feito é essencialmente para poder fazer testes da máquina sem precisar de um servidor SQL. Ao carregar no *radio button*, acontece que o SCADA faz um pedido ao PLC para alterar a fonte da receita dos pneus que são detetados no transportador de entrada.

A fonte só pode ser alterada pelo utilizador se a máquina estiver em modo manual. Quando a fonte de receitas está definida como "receita recebida do servidor SQL", a janela de cor branca torna-se invisível (o ecrã sem esta janela pode ser consultado na página 162). A janela a cinza mantém-se, e não é editável, permitindo ao operador saber a receita do pneu presente na estação de entrada.

Navegando na aba à esquerda, pode-se aceder ao ecrã com as definições de conexão - figura 4.32 e disponível na página 164. Este ecrã foi criado com o intuito

de permitir ao utilizador ter noção da arquitetura de rede da máquina e permitir ver se um dispositivo foi desligado, através da cor da borda de cada objeto: verde significa conectado e vermelho significa desconectado. Cada retângulo corresponde a um dispositivo, ao qual estão associados os vários possíveis IPs, para as várias redes em que se encontra.

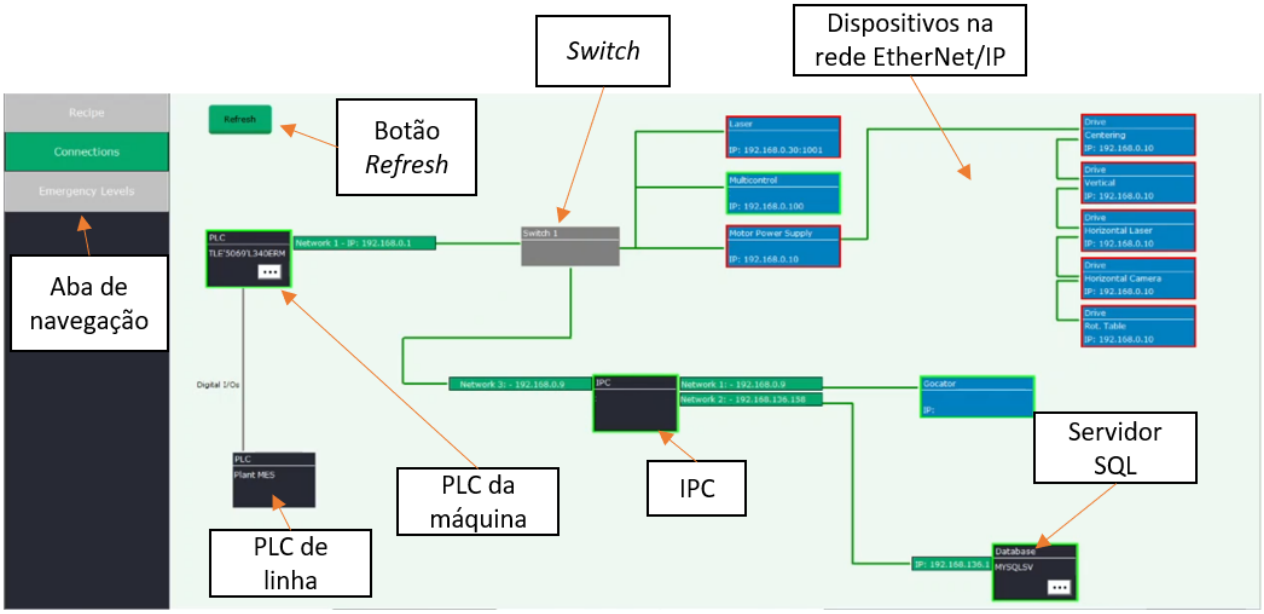


Figura 4.32: Ecrã *Seetings*, aba *Connections*.

Sempre que este ecrã é aberto, ou sempre que se carrega no botão *refresh* no canto superior esquerdo, corre um procedimento em VBA que envia um *ping* para estes dispositivos e avalia a sua resposta. Caso não haja resposta, significa que a conexão não está disponível e as bordas tomam o valor vermelho. Para outros dispositivos, tais como *drives* de servomotor e módulos de I/O, a verificação da conexão é feita pelo PLC através da sua comunicação por EtherNet/IP, e a cor das bordas corresponde a uma variável do PLC.

Para o PLC e para a base de dados SQL, pode-se ainda abrir uma pequena janela de controlo através do botão presente dentro do respetivo retângulo. A janela do PLC (página 165) contém informação sobre o seu estado atual e sobre as entradas/saídas trocadas com o PLC de linha.

A janela de controlo do servidor SQL (página 166) pode ser utilizada pelo operador para duas coisas. A primeira é saber qual o ID do pneu atualmente presente nas tabelas de entrada e saída, permitindo ao operador ter noção do que é que está presente no servidor SQL e detetar qualquer erro que tenha ocorrido. A segunda é eliminar cada uma das linhas desta tabela para corrigir os erros detetados.

Navegando na aba à esquerda para *emergency levels*, é aberto um ecrã que pode ser consultado na página 167. Este ecrã contém informação para o utilizador sobre o que acontece aos componentes para os dois níveis de emergência previstos na máquina. Este ecrã não tem nenhuma funcionalidade acrescentada, é somente informativo. Como não se trata de um ecrã crucial, não será aqui apresentado numa

figura.

4.5.5 Ecrã *Inputs & Outputs*

Como já foi explicado, este ecrã não foi criado porque à data da escrita, ainda não existia um conhecimento dos módulos de entradas e saídas que iriam ser colocados, nem dos componentes do quadro elétrico. Quando esta informação for adquirida, sugere-se que se faça uma organização semelhante à do ecrã *Settings*, com uma aba para navegar entre os possíveis quadros elétricos. Dentro de cada quadro, deverá ter informação sobre o sinal de todas as entradas e saídas da máquina.

4.5.6 Ecrã *Alarms & Events*

O último ecrã a apresentar permite fazer uma análise e gestão dos alarmes e eventos ativos. Isto é útil para o operador identificar os erros presentes nos vários componentes da máquina, ou para consultar o histórico de eventos que tenham ocorrido. Como se pode ver pela figura 4.33 (também disponível na página 168), utilizou-se uma *banner* predefinida do FactoryTalk.

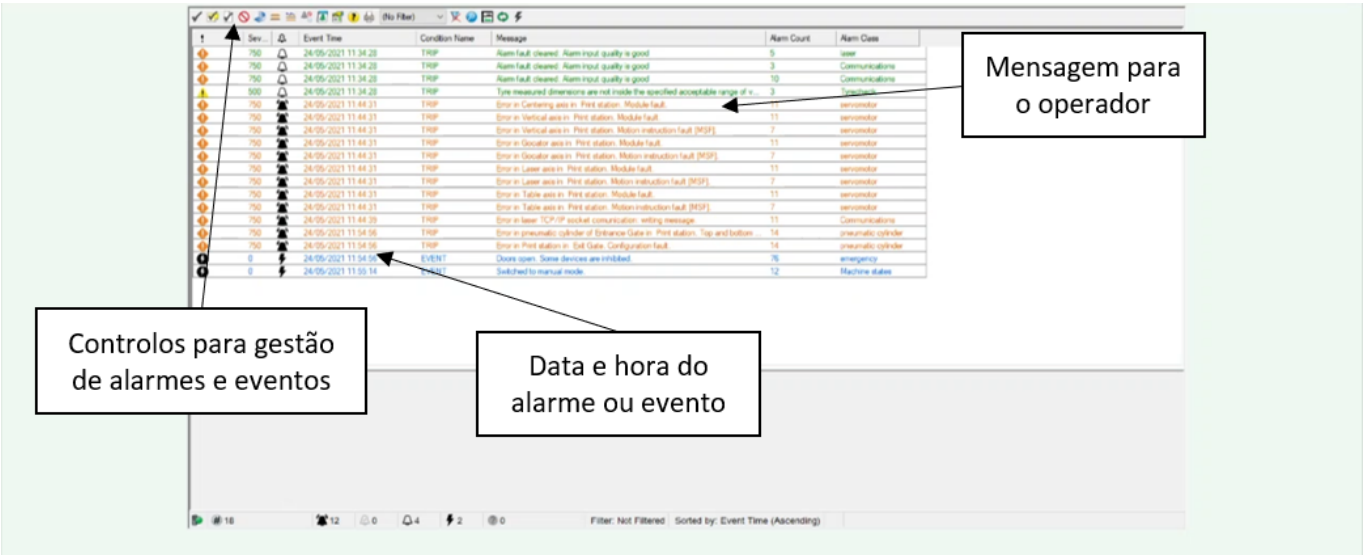


Figura 4.33: Ecrã *Alarms & Events*.

Os alarmes são uma funcionalidade do FactoryTalk configurados um a um pelo programador para serem acionados em função do valor de uma variável. A sua configuração tem imensa versatilidade porque é possível atribuir prioridades, classes e mensagens personalizadas a cada um. Adicionalmente, o FactoryTalk permite programar ações despoletadas pelo aparecimento de alarmes que poderiam vir a ser úteis, tais como geração de um relatório de erros.

Este objeto permite ainda arquivar e resolver alarmes, bem como organizá-los e filtrá-los por prioridade ou classe. Reconhece-se, no entanto, que esta *Banner* está desatualizada do ponto de vista estético, no que toca às cores e bordas utilizadas.

A maior parte dos alarmes configurados correspondem aos erros da máquina, referidos na secção 4.4 - Aplicação prática e testes. A diferença entre alarmes e eventos é que um evento não tem conotação negativa.

A título de exemplo, na figura referida, pode-se verificar que houve uma passagem do seletor manual/automático para a posição manual às 11:55:14 do dia 24/05/2021. Adicionalmente, verifica-se que estava ativo um alarme indicando que existia um erro em todos os módulos de eixo.

Termina assim a apresentação dos vários ecrãs que foram previstos para correr no SCADA desta máquina. Novamente, recomenda-se uma consulta do anexo C onde é possível ver em maiores dimensões todos os ecrãs criados.

4.6 Síntese do capítulo

Com o capítulo que aqui termina, foi apresentada ao leitor a programação desenvolvida no âmbito desta dissertação. Começando por introduzir alguns aspetos pertinentes acerca dos atuais PLCs da RA, seguiu-se para a apresentação de todos os componentes que requerem uma interface programada, e de que maneira esta foi concebida.

De seguida, foram apresentadas duas variáveis-chave do processo: o *status* do pneu e a sua receita. Para descrever o comportamento previsto da máquina no que toca a modos de funcionamento e rotinas, recorreu-se a seis *grafcets*: dois para controlo da máquina, um para a rotina de reinicialização e três para os ciclos de modo automático de cada estação.

Estando o comportamento da máquina delineado, apresentaram-se os modelos práticos utilizados para testar o código feito, que serviram como forma de validação de um dos objetivos desta dissertação - a programação do PLC. Dedicou-se ainda uma subsecção à explicação da organização e implementação do código dentro do Studio 5000.

Em último lugar, foram apresentados os ecrãs criados para correr na visualização do FactoryTalk. Através de *printscreens* dos ecrãs principais, foram referidos e explicados os elementos presentes e funcionalidades implementadas em cada um. Assim, conclui-se o capítulo de programação da máquina e com ele o corpo desta dissertação.

Capítulo 5

Conclusão e trabalhos futuros

Chega-se assim ao capítulo final desta dissertação. Aqui será feito um apanhado geral daquilo que foi coberto ao longo dos vários capítulos, recordando os tópicos principais de cada um. Finalmente, haverá uma secção de trabalhos futuros por onde se poderia prosseguir para finalizar ou melhorar alguns aspetos da máquina.

5.1 Conclusão

A seleção de componentes de automação passa inicialmente por um estudo dos fornecedores existentes e das suas gamas orientadas às diferentes aplicações. Para poder comparar soluções, é necessário saber que métricas usar e sob que condições, para que a escolha final seja justa e a melhor para a aplicação. Por outro lado, o sobredimensionamento não é uma prática surpreendente e por vezes até é desejável, como garantia do alcance de objetivos: nestas ocasiões é melhor errar por excesso do que por defeito.

No âmbito desta dissertação, todos os componentes selecionados foram alvo de uma análise cuidadosa e escolhidos justificadamente. Foram abordados servomotores, caixas redutoras, elementos do circuito elétrico de potência, dispositivos de controlo da máquina e vários sensores de diferentes marcas. O passo seguinte seria proceder ao contacto com os fornecedores: só através do preço, que não foi aqui considerado, se poderá obter a solução final.

A etapa de programação do controlador de uma máquina requer, acima de tudo, organização e método. A divisão do programa em partes funcionais e a definição de como será hierarquizado o controlo, permitem criar fundações sólidas sob as quais poderá ser implementado qualquer funcionamento pretendido. Este planeamento é feito independentemente da marca de controladores utilizados e representado recorrendo a diagramas temporais e *grafcets*.

Para poder programar o comando dos demais componentes da máquina, é necessário estudá-los previamente para garantir uma interface sólida e suave. A melhor fonte de conhecimento é o manual do próprio equipamento, mas a discussão com outros profissionais e a consulta da *Internet* são sempre uma mais-valia. No âmbito prático, é comum encontrar em fóruns *online* indivíduos com problemas de caráter semelhante àqueles que se encontram ao longo da programação; em contrapartida,

a desinformação também não é de todo censurada.

Apenas planeando e testando é possível construir soluções robustas. Na falta de um meio de teste, é necessário recorrer a simuladores ou criá-los para que se possa validar trabalho. A solução de programação que aqui se apresenta foi devidamente validada recorrendo a uma banca de teste disponibilizada pela empresa em conjunto com vários simuladores, pelo que se encontra pronta para passar à fase de implementação e arranques iniciais de uma máquina real.

A programação, que foi levada a cabo no *software* Studio 5000 da Rockwell Automation, passou pela criação de:

- Protocolos para fluxo de informação e material;
- Uma hierarquia para controlo do funcionamento das estações;
- Rotinas de processo da máquina e de recuperação pós-falha;
- Interfaces com os componentes da máquina;
- Mecanismos de deteção e gestão de situações de erro.

A visualização a decorrer na interface homem-máquina é um aspeto fulcral do produto final porque deve transparecer toda a qualidade do projeto feito até então. O funcionamento e arquitetura da máquina devem ser subentendidos através da navegação nos ecrãs, dando ao operador a informação necessária para que possa fazer diagnóstico, comando e parametrização. O ambiente de desenvolvimento utilizado para criação de ecrãs - o FactoryTalk - é altamente funcional e as suas capacidades estendem-se muito para além daquelas que foram necessárias implementar.

Nesse sentido, foi desenvolvida toda uma visualização, dividida em grupos funcionais, com todas as informações que o operador poderá precisar. A navegabilidade entre ecrãs e a centralização de informação foram as chaves para alcançar o resultado final, que se apresentou com recurso a *printscreens* tirados durante a fase final do desenvolvimento.

Em termos de balanço pessoal, o autor considera que o tema desta dissertação se insere perfeitamente na sequência do mestrado em automação industrial. As partes que se revelaram mais complicadas foram o planeamento do protocolo de transferência de um pneu entre a máquina e o transportador da fábrica, e a criação de ecrãs do SCADA. Depois da experiência na empresa, o autor considera-se orientado no sentido de expandir os seus conhecimentos nas áreas que dizem respeito a AGVs, robótica, fluídos de potência (pneumática e óleo-hidráulica), comissionamento de eixos servomotorizados e segurança de máquinas.

5.2 Trabalhos Futuros

Visto que a máquina ainda não está construída, ainda existe bastante trabalho a realizar antes que possa ser integrada devidamente numa fábrica. É ainda importante distinguir entre trabalhos futuros no contexto desta dissertação e no contexto de outras aplicações. Cada cliente tem a sua necessidade específica, pelo que a máquina sofre sempre certas alterações em função daquilo que é encomendado.

No contexto desta dissertação, os trabalhos futuros podem ser apresentados em tópicos:

- Projetar o sistema de controlo de segurança segundo a ISO 13849-1;
- Proceder à compra dos componentes selecionados, em função do preço de venda e prazos de entrega;
- Integração da câmara de visão;
- Adição de cartas de entradas e saídas do PLC e programação dos devidos sinais dos sensores e atuadores;
- Completar o programa do PLC com sinais de monitorização de vários componentes do quadro elétrico;
- Completar as rotinas de *interlock* da máquina;
- Completar o ecrã da HMI que diz respeito aos sinais dos sensores lidos;
- Ajustar os parâmetros máquina devidamente preparados no *software*;
- Fazer o devido comissionamento dos eixos servomotorizados;
- Proceder a uma fase de testes da máquina para provar o conceito.

Após uma abordagem de todos estes tópicos, chegar-se-ia à solução chave-na-mão.

Referências bibliográficas

[1] Wikipedia. Pneu, 2021. URL https://pt.wikipedia.org/w/index.php?title=Pneu&oldid=60246969.

[2] Desconhecido. Robert william thomson, 2007. URL https://web.archive.org/web/20070929200017/http://www.mearns.org.uk/stonehaven/thomson.htm.

[3] Bridgestone. Materials, n.d. URL https://www.bridgestone.com/products/basic_knowledge/materials/.

[4] Allied Market Research. Automotive tire market statistics - 2027, 2020. URL https://www.alliedmarketresearch.com/automotive-tire-market.

[5] ETRMA. Etrma members' tyre sales in europe: Covid 19 strongly affected the market in 2020, 2021. URL https://www.etrma.org/wp-content/uploads/2021/01/20210121_Tyre-sales-2020_PR1.pdf.

[6] Goodyear. Understanding your tyre, 2020. URL https://www.goodyear.eu/en_za/consumer/learn/how-tires-are-made.html.

[7] KBA Mechatronic. Laser spa series c-10 & c-30 user manual, 2017.

[8] Rockwell Automation. Mp-series low-inertia servo motor with 100 mm to 165 mm frame sizes, 2018. URL https://literature.rockwellautomation.com/idc/groups/literature/documents/in/mp-in001_-en-p.pdf.

[9] Rockwell Automation. Rotary motors, 2021. URL https://www.rockwellautomation.com/en-us/products/hardware/allen-bradley/motion-control/servo-motors/rotary-motors.html.

[10] EESCO. Rockwell automation #2090-cfbm7df-cdaf05 servo drive cable assembly motor feedback cable, 2021. URL https://buy.eescodist.com/Automation-and-Control/Drives/Encoder-Cables/ROCKWELL-AUTOMATION/Servo-Drive-Cable-Assembly-Motor-Feedback-Cable/2090-CFBM7DF-CDAF05/p/88495112095-1?Ntt=%232090%5E*.

[11] EESCO. Rockwell automation #2090-csbm1df-14aa05 servo drive cable assembly motor power cable w/brake wires, 2021. URL https://buy.eescodist.com/Automation-and-Control/

Drives/Encoder-Cables/ROCKWELL-AUTOMATION/
Servo-Drive-Cable-Assembly-Motor-Power-Cable-w-Brake-Wires/
2090-CSBM1DF-14AA05/p/88495126789-1?Ntt=Servo+Drive+Cable+
Assembly+Motor+Power+w%2FBrake+Wires%2C+5m.

[12] EESCO. Rockwell automation #2090-csbm1e1-14af05 single motor power cable, 5 meter l, 2021. URL https://buy.eescodist.com/Automation-and-Control/Drives/Encoder-Cables/ROCKWELL-AUTOMATION/Single-Motor-Power-Cable-5-Meter-L/2090-CSBM1E1-14AF05/p/88950809987-1?Ntt=Single+motor+power+cable.

[13] Siemens. Simotics - drive technology - 1fk7 g2 synchronous motors, 2018. URL https://support.industry.siemens.com/cs/document/55379345/simotics-s-1fk7-g2-synchronous-motors?dti=0&dl=en&pnid=13349&lc=pt-BR.

[14] Wittenstein Alpha. alpha basic line & alpha value line product catalog, 2020.

[15] Kristin Lewotsky. Understanding the mysteries of inertia mismatch, 2015. URL https://www.automate.org/industry-insights/understanding-the-mysteries-of-inertia-mismatch.

[16] Siemens. Catalog d 21.4 - sinamics s120 and simotics, 2017.

[17] Rockwell Automation. Mp-series low inertia servo motors product profile, 2012.

[18] Siemens. Sinamics low voltage engineering manual, 2020.

[19] Rockwell Automation. Kinetix 5700 servo drives user manual, 2020.

[20] LMI Technologies. Gocator displacement sensor user manual, 2019.

[21] LMI Technologies. Maximum encoder rate, 2020. URL https://support.lmi3d.com/hc/en-us/articles/360033662151-Maximum-encoder-rate.

[22] Sick. Dfs60 product family overview, 2021. URL https://cdn.sick.com/media/familyoverview/8/28/428/familyOverview_DFS60_g244428_en.pdf.

[23] Electro Sensors. Gv204, 2021. URL https://www.electro-sensors.com/motrona-products/motrona-interface/gv204.

[24] Rockwell Automation. Encoder output emulator installation instructions, 2016. URL https://literature.rockwellautomation.com/idc/groups/literature/documents/in/2198-in013_-en-p.pdf.

[25] PLC City. 2198-abqe allen-bradley, 2021. URL https://www.plc-city.com/shop/en/rockwell/rck-2198-abqe-nfs.html.

[26] M.H. Schwarz and J. Borcsok. A survey on opc and opc-ua: About the standard, developments and investigations. pages 1–6, 10 2013. doi: 10.1109/ICAT.2013.6684065.

[27] Advantech. Aiis-3400p (poe), 2020. URL https://advdownload.advantech.com/productfile/PIS/AIIS-3400/file/AIIS-3400P_U_DS(071420)20201026103006.pdf.

[28] Microsoft. Introducing windows 10 editions, 2015. URL https://blogs.windows.com/windowsexperience/2015/05/13/introducing-windows-10-editions/.

[29] Rockwell Automation. Industrial computers and monitors specifications technical data, 2020. URL https://literature.rockwellautomation.com/idc/groups/literature/documents/td/ic-td001_-en-p.pdf.

[30] Blackbox. Dvi-d digital video extender kit, 2021. URL https://www.blackbox.be/en-be/fi/1473/12560/DVI-D-Digital-Video-Extender-Kit/.

[31] Blackbox. Dual-link dvi repeater, 2021. URL https://www.blackbox.be/en-be/fi/1473/13747/Dual-Link-DVI-Repeater/.

[32] Omron. Technical explanation for photoelectric sensors, 2021. URL https://www.ia.omron.com/data_pdf/guide/43/photoelectric_tg_e_8_4.pdf.

[33] Sick. Dual lens concept, 2021. URL https://www.sick.com/fi/en/glossary/dual-lens-concept/g/p500853.

[34] Sick. Autocollimation, 2021. URL https://www.sick.com/ag/en/glossary/autocollimation/g/p500844.

[35] Sick. Small photoelectric sensors g10, 2021. URL https://www.sick.com/ag/en/p/p580846.

[36] Sick. Measuring automation light grids mlg-2, 2021. URL https://www.sick.com/ag/en/automation-light-grids/measuring-automation-light-grids/mlg-2/mlg02n-0445h13201/p/p514786.

[37] Wikipédia. Clepsidra, 2021. URL https://pt.wikipedia.org/wiki/Clepsidra.

[38] PLC DEV. The birth of the plc, 2009. URL http://www.plcdev.com/the_birth_of_the_plc.

[39] Dummermuth. Allen bradley plc patent #3,942,158, 1976. URL http://www.plcdev.com/allen_bradley_plc_patent_3942158_by_dummermuth.

[40] Wikipedia. Rockwell automation, 2021. URL https://en.wikipedia.org/wiki/Rockwell_Automation.

[41] BIN 95. Plc pac difference - pac automation controller defined, 2021. URL https://bin95.com/articles/automation/plc-pac-difference.htm.

[42] B.R. Mehta and Y.J. Reddy. Chapter 8 - programmable automation controller. In B.R. Mehta and Y.J. Reddy, editors, *Industrial Process Automation Systems*, pages 301–306. Butterworth-Heinemann, Oxford, 2015. ISBN 978-0-12-800939-0. doi: https://doi.org/10.1016/B978-0-12-800939-0.00008-5. URL https://www.sciencedirect.com/science/article/pii/B9780128009390000085.

[43] Austin SCOTT. Learning rslogix 5000 programming, 2020.

[44] Industry Insights. Understanding motion control networks, 2016. URL https://www.automate.org/industry-insights/understanding-motion-control-networks.

[45] Real Time Automation. Ethernet/ip and tcp/ip, 2016. URL https://www.rtautomation.com/rtas-blog/ethernet-ip-and-tcp-ip/.

[46] KBA Mechatronic. Laser system / scanlinux - interface protocol, 2015.

[47] Rockwell Automation. Ethernet/ip socket interface application technique, 2020.

[48] Interrol. Supplement start-up and programming multicontrol ai/bi, rollerdrive ec5000 bi, 2020.

[49] Microsoft. Dynamic link library (dll), 2020. URL https://docs.microsoft.com/en-us/troubleshoot/windows-client/deployment/dynamic-link-library.

[50] Wikipedia. Activex data objects, 2021. URL https://en.wikipedia.org/wiki/ActiveX_Data_Objects.

[51] Wikipedia. Ole db - object linking and embedding, 2021. URL https://en.wikipedia.org/wiki/OLE_DB.

[52] Wikipedia. Com - component object model, 2021. URL https://en.wikipedia.org/wiki/Component_Object_Model.

[53] Rockwell Automation. Logix 5000 controllers, 2009. URL https://literature.rockwellautomation.com/idc/groups/literature/documents/qs/1756-qs001_-en-p.pdf.

Todas as referências com página web se encontravam disponíveis a 28 de junho de 2021, data de entrega da versão provisória da dissertação.

Anexo A

Cálculos realizados para a seleção de servomotores e caixas redutoras

Neste anexo serão apresentados todos os passos seguidos para efetuar a seleção dos servomotores e caixas redutoras para uma solução Siemens e uma solução em RA. Recomenda-se a leitura prévia da secção 3.1.1 - Seleção de servomotores e caixas redutoras para uma melhor compreensão do assunto aqui tratado.

Eixo A

O servomotor A está projetado para ser acoplado a uma caixa redutora com saída por veio, que por sua vez está a acionar uma polia (figura A.1). A ligação entre a saída da caixa e a polia não é direta (ou seja, o veio de saída da redutora não engrena na polia), é pois feita através de um acoplamento entre ambas. Este tipo de ligações permite uma fácil montagem, manutenção e ampla gama de compatibilidade de motores para acionar o movimento de centragem, e está protegida por um corpo de revolução com ligação aparafusada ao redutor para proteção da ligação.

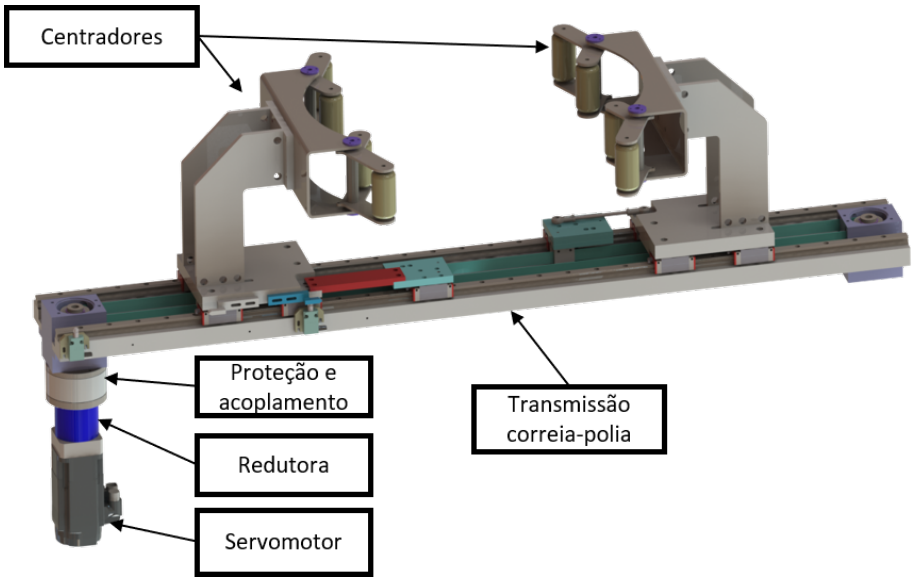


Figura A.1: Componentes que constituem o eixo A. Projeto: Gislotica.

A polia acionada faz movimentar uma correia. Acopladas a cada um dos lados desta correia estão duas carruagens em forma de pescoço de cavalo que se movem na mesma direção, mas em sentidos opostos provocando o arrasto do pneu para uma posição centrada no tapete garantindo fisicamente que ambas se movem à mesma velocidade. Este acionamento, a pedido do cliente, também está preparado para ser feito por um cilindro pneumático através do arrasto linear de uma das carruagens, que por sua vez arrasta a correia, que por sua vez arrasta a outra carruagem.

A solução servomotorizada tem vantagens no sentido de permitir controlo facilitado por software, posicionamento mais preciso e maior capacidade dinâmica. Por outro lado, a solução pneumática pode revelar-se mais barata e compacta.

Os requisitos dinâmicos são resultado da análise do ciclo de trabalho deste servomotor. Para efeitos de cálculo, considerar-se-á o seguinte ciclo: 1 s para avançar e 1 s para recuar (tempo de trabalho total de 2 s), com DC = 0,5 (tempo de ciclo total 4 s). Será assumido que o eixo percorre toda a distância do seu curso (350 mm) durante o movimento, que o pneu começa logo a ser empurrado por uma das carruagens (cenário mais crítico em termos de binário) e um tempo de aceleração e desaceleração de 100 ms. A massa da carga é 50 kg, e a do pneu majorada em 50 kg.

De acordo com a experiência da empresa, deve-se prever uma força de atrito devido ao arrasto do pneu de cerca de 300 N, ou seja, um coeficiente de atrito

constante de cerca de $\mu = 0.6$. À partida, este valor para o coeficiente de atrito pode soar demasiado alto para o contacto borracha-metal que existe entre o pneu e o rolo do transportador, contudo é preciso considerar que ao fim de muitos ciclos, os rolos ficam sujos com vestígios de borracha, e o contacto passa a ser borracha-borracha levando a um aumento deste coeficiente.

Sabe-se que a polia tem um diâmetro de 57 mm e assume-se um rendimento do mecanismo correia-polia (incluindo guias) de 80 %. As curvas de posição e velocidade para o movimento de avanço são apresentadas na figura A.2.

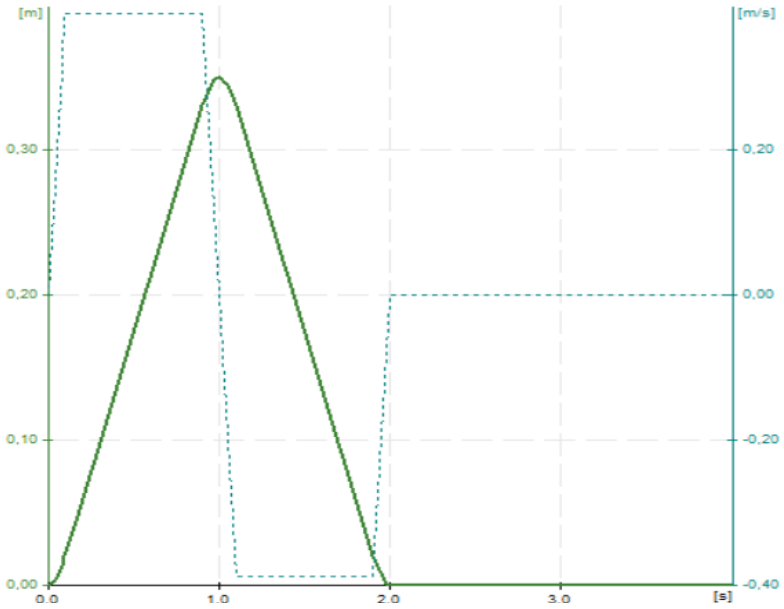


Figura A.2: Perfil temporal de posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo A. Fonte: SIZER da Siemens.

Com os valores conhecidos substituídos, pode-se calcular a distância percorrida através da equação 3.1:

$$0,35 = \int_{0}^{0,1} (\frac{v_{const}}{0,1} \times t)\ dt + v_{const} \times (1 - 0,1 - 0,1) + \int_{1-0,1}^{1} (v_{const} - \frac{v_{const}}{0,1} \times t)\ dt$$

Integrando:

$$0,35 = v_{const} \times 0,05 + v_{const} \times 0,8 + v_{const} \times 0,05$$

Resolvendo em ordem a v_{const}:

$$v_{const} = 0,389\ m/s$$

A aceleração - $a_{inicial}$, que é igual à desaceleração, é calculada usando a equação 3.2:

$$a_{inicial} = \frac{0,389}{0,1} = 3,89\ m/s^2$$

O perfil de binário - T_{polia} e velocidade angular em rpm - n_{polia} na polia durante o ciclo considerado estão representados na figura A.3.

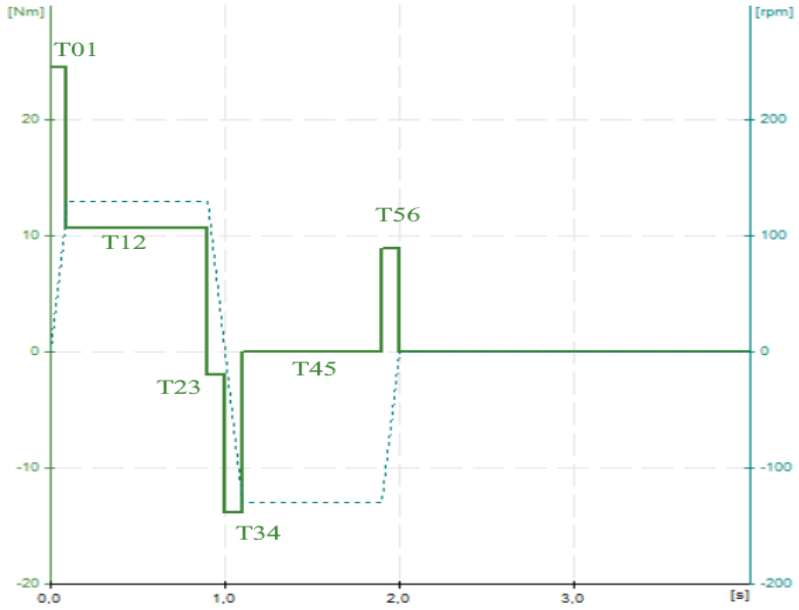


Figura A.3: Perfil temporal do binário - T_{polia} (traço contínuo) e velocidade angular - ω_{polia} (traço interrompido) da polia do eixo A. Fonte: SIZER da Siemens.

Note-se que ao longo do ciclo, a variável T_{polia} pode tomar seis valores distintos, cujo significado é apresentado abaixo. Relembra-se que o pneu só é arrastado no avanço.

- $T_{0\text{-}1}$ corresponde ao arranque no avanço;
- $T_{1\text{-}2}$ corresponde ao avanço em movimento uniforme;
- $T_{2\text{-}3}$ corresponde à frenagem no avanço;
- $T_{3\text{-}4}$ corresponde ao arranque no recuo;
- $T_{4\text{-}5}$ corresponde ao recuo em movimento uniforme;
- $T_{5\text{-}6}$ corresponde à frenagem no recuo.

O valor do binário exercido na polia ao longo do ciclo é calculado através das equações 3.4 e 3.3:

$$T_{polia\ -\ motor} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \frac{1}{\eta_{c\text{-}p}} \times \frac{\phi_{polia}}{2}$$

$$T_{polia\ -\ gerador} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \eta_{c\text{-}p} \times \frac{\phi_{polia}}{2}$$

$$T_{0\text{-}1\ -\ polia} = (100 \times 3,89 + 300) \times \frac{1}{0,8} \times \frac{0,057}{2} = 24,54\ N.m$$

$$T_{1\text{-}2\ -\ polia} = (100 \times 0 + 300) \times \frac{1}{0,8} \times \frac{0,057}{2} = 10,69\ N.m$$

$$T_{2\text{-}3\ -\ polia} = (100 \times (-3,89) + 300) \times 0,8 \times \frac{0,057}{2} = -2,03\ N.m$$

$$T_{3\text{-}4\ -\ polia} = (50 \times (-3,89) + 0) \times \frac{1}{0,8} \times \frac{0,057}{2} = -6,93\ N.m$$

$$T_{4\text{-}5\ -\ polia} = (50 \times 0 + 0) \times \frac{1}{0,8} \times \frac{0,057}{2} = 0,00\ N.m$$

$$T_{5\text{-}6\ -\ polia} = (50 \times 3,89 + 0) \times 0,8 \times \frac{0,057}{2} = 4,43\ N.m$$

Consultando [54], encontra-se o modelo de redutores da gama NP que mais se adequa a esta aplicação (modelo NP 015 MF2). Considerando que a velocidade de rotação do servomotor deve ser elevada para que a sua eficiência seja também elevada, procura-se qual a razão de transmissão - $i_{redutora\text{-}aprox}$ que coloca o servomotor na zona de movimento uniforme perto da velocidade nominal média de entrada da caixa redutora.

No catálogo apresentam-se velocidades médias permitidas de entrada da caixa redutora entre 3800 rpm e 4000 rpm, para 20 ºC. Assumindo que a temperatura será maior que 20 ºC, será assumida uma velocidade média de entrada de 3500 rpm. Procura-se então qual é o $i_{redutora\text{-}aprox}$ que se pode ter, para não exceder a velocidade de rotação de entrada de 3500 rpm:

$$i_{redutora\text{-}aprox} = \frac{3500}{\omega_{polia} \times \frac{60}{2\pi}}$$

$$\omega_{polia} = \frac{v_{const}}{\frac{\phi_{polia}}{2}}$$

$$\omega_{polia} = \frac{0,389}{\frac{0,057}{2}} = 13,65\ rad/s$$

$$i_{redutora\text{-}aprox} = \frac{3500}{13,65 \times \frac{60}{2\pi}} = 26,85$$

O valor de razão de transmissão selecionado é $i_{redutora} = 20$ e as características pertinentes da caixa escolhida encontram-se resumidas na tabela A.1. Verifica-se que o binário de aceleração permitido nesta caixa é superior a $T_{0\text{-}1}$ e o binário máximo permitido é mais de 1,5 vezes superior a $T_{0\text{-}1}$. O diâmetro do cubo de entrada só pode ser escolhido aquando da seleção do servomotor, mas adianta-se já que se escolherá o de 9 mm.

Tabela A.1: Características técnicas da caixa redutora escolhida para o eixo A.

Modelo	Binário máx. aceleração (N.m)	Binário máx. permitido (N.m)	$i_{redutora}$	$J_{redutora}$ (kg.cm^2)	$\eta_{redutora}$ (%)
NP 015s-MF2	35	$50/1,5 = 34$	20	0,03	90

Estando a caixa redutora selecionada, podem-se calcular os binários $T_{0\text{-}1\ -\ servo}$ até $T_{5\text{-}6\ -\ servo}$. Desprezando, numa primeira instância, o binário devido às forças de inércia necessário para acelerar o veio do servomotor ($J_{servo} = 0\ kg.m^2$, porque ainda não se escolheu um servomotor), utilizam-se as equações 3.6, 3.7 e 3.8.

$$T_{servo\ -\ motor} = (\frac{T_{polia\ -\ motor}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \frac{1}{\eta_{redutora}}$$

$$T_{servo\ -\ gerador} = (\frac{T_{polia\ -\ gerador}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \eta_{redutora}$$

$$T_{0\text{-}1\ -\ servo} = (\frac{24,54}{20} + 0,000003 \times \frac{13,65 \times 25}{0,1}) \times \frac{1}{0,9} = 1,37\ N.m$$

$$T_{1\text{-}2\ -\ servo} = (\frac{10,69}{20} + 0,000003 \times 0) \times \frac{1}{0,9} = 0,59\ N.m$$

$$T_{2\text{-}3\ -\ servo} = (\frac{-2,03}{20} + 0,000003 \times \frac{-13,65 \times 25}{0,1}) \times 0,9 = -0.10\ N.m$$

$$T_{3\text{-}4\ -\ servo} = (\frac{-6,93}{20} + 0,000003 \times \frac{-13,65 \times 25}{0,1}) \times \frac{1}{0,9} = -0,40\ N.m$$

$$T_{4\text{-}5\ -\ servo} = (\frac{0}{20} + 0,000003 \times 0) \times \frac{1}{0,9} = 0,00\ N.m$$

$$T_{5\text{-}6\ -\ servo} = (\frac{4,43}{20} + 0,000003 \times \frac{13,65 \times 25}{0,1}) \times 0,9 = 0,21\ N.m$$

O binário efetivo é calculado segundo a equação 3.9:

$$T_{eff} = \left(\frac{0,1 \times 1,37^2 + 0,8 \times 0,59^2 + 0,1 \times 0.10^2 + 0,1 + 0,8 + 0,1 + \frac{0,1 \times 0,78^2 + 0,8 \times 0 + 0,1 \times 0,41^2 + 2 \times 0}{0,1 + 0,8 + 0,1 + 2} \right)^{1/2}$$

$$T_{eff} = 0,35\ N.m$$

Começando pela marca Siemens, o servomotor escolhido foi o 1FK7022-5AK7, cujas características podem ser consultadas na tabela A.2. O binário médio efetivo está a 52 % do limite recomendado, o binário máximo está a 43 % do limite e não se está a atingir a velocidade nominal do servomotor.

A razão de inércias é dada pelas equações 3.10 e 3.11, e é calculada para o caso em que a carga tem uma massa de 100 kg (avanço):

Tabela A.2: Características técnicas do servomotor Siemens escolhido para o eixo A.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade nominal (rpm)	J_{servo} (kg.cm^2)
1FK7022-5AK7	3,40	0,60	6000	0,35

$$\frac{1}{2} \times J_{eq} \times {\omega_{servo}}^2 = \frac{1}{2} \times 100 \times \left(\frac{\omega_{servo} \times \frac{0{,}057}{2}}{20} \right)^2 + \frac{1}{2} \times 0,000003 \times {\omega_{servo}}^2$$

$$J_{eq} = 2,06 \ kg.cm^2$$
$$r_{inércias} = \frac{2,06}{0,35} = 5,89$$

O valor calculado da razão de inércias adequa-se e está dentro do intervalo predefinido. Dado que este eixo não tem necessidade de posicionamento preciso, o valor da razão de inércias poderia ser mais elevado.

Fica assim selecionado o servomotor da marca Siemens para o eixo A. Note-se que apesar de o texto ter levado o leitor diretamente a uma combinação servomotor/caixa redutora, durante a seleção fizeram-se várias iterações, sendo esta o resultado final.

Do lado da marca RA, o servomotor escolhido foi o MPL-B1530U, cujas características podem ser consultadas na tabela A.3. O binário médio efetivo está a 41 % do limite recomendado, o binário máximo está a 44 % do limite e não se está a atingir a velocidade máxima do servomotor.

Tabela A.3: Características técnicas do servomotor Rockwell Automation escolhido para o eixo A.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade máxima (rpm)	J_{servo} (kg.cm^2)
MPL-B1530U	2,82	0,6	7000	0,35

A razão de inércias é dada pela equação 3.10 e 3.11, e é calculada para o caso em que a carga tem uma massa de 100 kg (avanço):

$$J_{eq} = 1,33 \ kg.cm^2$$
$$r_{inércias} = \frac{1,33}{0,23} = 5,8$$

Neste caso, verifica-se que, apesar de o servomotor estar ligeiramente folgado, o imediatamente abaixo (MPL-B1520U) fica com uma razão de inércias superior a 10, pelo que não será escolhido. Fica assim selecionado o servomotor da RA para o eixo A. Assim, dá-se como concluída a seleção de servomotores para este eixo.

Eixo B

O servomotor B, tal como o servomotor A, está projetado para ser acoplado a uma caixa redutora com saída por veio, que por sua vez está a acionar duas polias de duas transmissões polia-correia, através de um veio de ligação entre elas (figura 2.4). Este conjunto permite movimentação vertical do LASER e da câmara ao mesmo tempo. Neste caso, ao contrário do eixo anterior, é necessário maior rigor de posicionamento, pelo que a solução não poderia ser pneumática.

O ciclo de trabalho deste motor consiste num avanço e recuo, com duração de 2 s cada (tempo de trabalho total de 4 s), e $DC = 0,5$ o que resulta num tempo total de ciclo de 8 s.

Será assumido que o eixo percorre toda a distância do seu curso (750 mm) durante o movimento e um tempo de aceleração e desaceleração de 100 ms. A massa total da carga é 200 kg, a polia tem um diâmetro de 57 mm e assume-se um rendimento total de ambos os mecanismos correia-polia (incluindo guias) de 80 %. As curvas de posição e velocidade para o movimento de avanço são apresentadas na figura A.4.

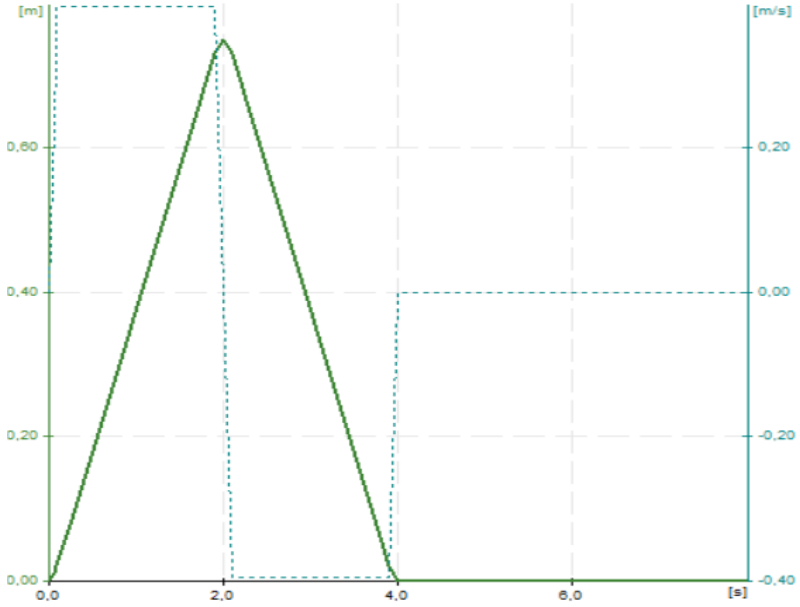


Figura A.4: Perfil temporal da posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo B. Fonte: SIZER da Siemens.

Com os valores conhecidos substituídos, pode-se calcular a distância percorrida através da equação 3.1:

$$0,75 = \int_{0}^{0,1} (\frac{v_{const}}{0,1} \times t)\ dt + v_{const} \times (1,8 - 0,1 - 0,1) + \int_{2-0,1}^{2} (v_{const} - \frac{v_{const}}{0,1} \times t)\ dt$$

Integrando:

$$0,75 = v_{const} \times 0,05 + v_{const} \times 1,8 + v_{const} \times 0,05$$

Resolvendo em ordem a v_{const}:

$$v_{const} = 0,395\ m/s$$

A aceleração inicial - $a_{inicial}$, que é igual à desaceleração, é calculada usando a equação 3.2:

$$a_{inicial} = \frac{0.395}{0.1} = 3,95\ m/s^2$$

Apesar de neste eixo haver duas calhas, para efeitos de dimensionamento do servomotor, considerou-se que apenas existe uma a segurar o dobro da carga. Como tal, as contas podem ser feitas com uma ordem semelhante àquela que foi feita no eixo A. Note-se que se trata de um eixo vertical e, portanto, na paragem durante o ciclo, o motor está na mesma a produzir binário para segurar a carga.

Apesar de o servomotor ter um travão, este não deve ser utilizado como travão durante o ciclo, apenas para a situação de falta de energia no drive. Mesmo no caso de uma paragem de emergência, não se deve recorrer a este travão: uma alternativa é comandar o servomotor a parar antes de lhe retirar, através da utilização de um relé de segurança temporizado que desenergiza a saída do *drive* através do *Safe Torque Off* (STO) passado algumas centenas de milissegundos, dando tempo suficiente para executar um comando de imobilização do servomotor.

O perfil de binário - T_{polia} e velocidade angular em rpm - n_{polia} na polia durante o ciclo considerado estão representados na figura A.5.

Note-se que ao longo do ciclo, a variável T_{polia} pode tomar sete valores distintos, cujo significado é apresentado abaixo.

- $T_{0\text{-}1}$ corresponde ao arranque no avanço;
- $T_{1\text{-}2}$ corresponde ao avanço em movimento uniforme;
- $T_{2\text{-}3}$ corresponde à frenagem no avanço;
- $T_{3\text{-}4}$ corresponde ao arranque no recuo;
- $T_{4\text{-}5}$ corresponde ao recuo em movimento uniforme;
- $T_{5\text{-}6}$ corresponde à frenagem no recuo.
- $T_{6\text{-}7}$ corresponde à paragem, segurando a carga.

O valor destes binário é calculado através das equações 3.4 e 3.3:

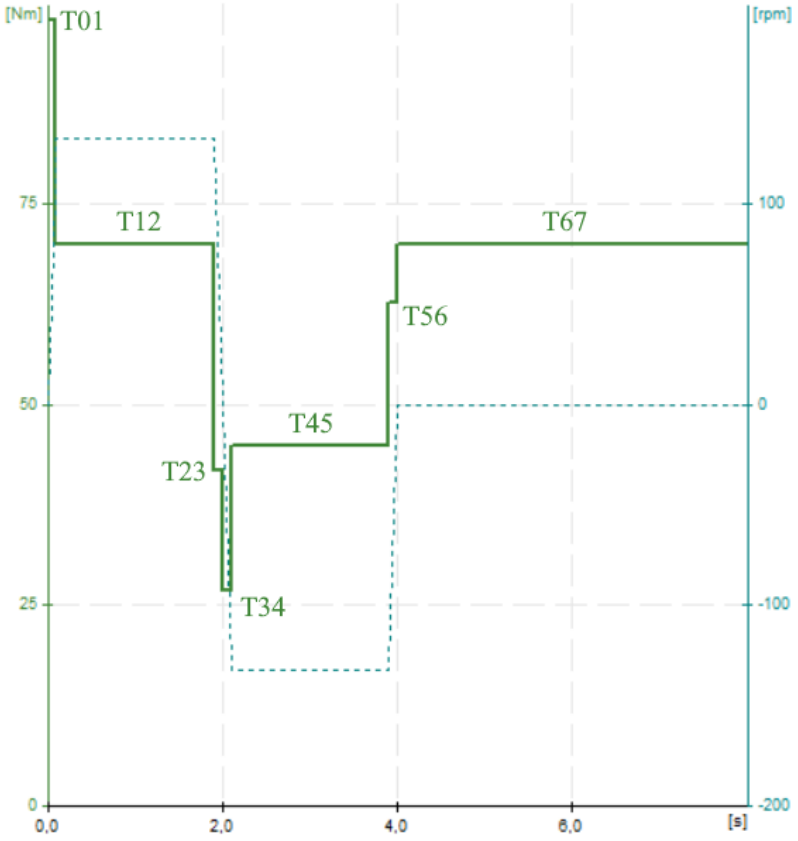


Figura A.5: Perfil temporal do binário - T_{polia} (traço contínuo) e velocidade angular - ω_{polia} (traço interrompido) na polia do eixo B. Fonte: SIZER da Siemens.

$$T_{polia\ -\ motor} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \frac{1}{\eta_{c\text{-}p}} \times \frac{\phi_{polia}}{2}$$

$$T_{polia\ -\ gerador} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \eta_{c\text{-}p} \times \frac{\phi_{polia}}{2}$$

$$T_{0\text{-}1\ -\ polia} = (200 \times 3,95 + 200 \times 9,81) \times \frac{1}{0,8} \times \frac{0,057}{2} = 98,04\ N.m$$

$$T_{1\text{-}2\ -\ polia} = (200 \times 0 + 200 \times 9,81) \times \frac{1}{0,8} \times \frac{0,057}{2} = 69,87\ N.m$$

$$T_{2\text{-}3\ -\ polia} = (200 \times (-3,95) + 200 \times 9,81) \times \frac{1}{0,8} \times \frac{0,057}{2} = 41,75\ N.m$$

$$T_{3\text{-}4\ -\ polia} = (200 \times (-3,95) + 200 \times 9,81) \times 0,8 \times \frac{0,057}{2} = 26,72\ N.m$$

$$T_{4\text{-}5\ -\ polia} = (200 \times 0 + 200 \times 9,81) \times 0,8 \times \frac{0,057}{2} = 44,72\ N.m$$

$$T_{5\text{-}6\ -\ polia} = (200 \times 3,95 + 200 \times 9,81) \times 0,8 \times \frac{0,057}{2} = 62,75\ N.m$$

$$T_{6\text{-}7\ -\ polia} = (200 \times 0 + 200 \times 9,81) \times \frac{1}{0,8} \times \frac{0,057}{2} = 69,87\ N.m$$

Perante isto, poder-se-ia arguir que o servomotor está a produzir binário des-

necessariamente nas secções 2-3 e 3-4. Ou seja, apenas com as forças do peso, era possível adquirir uma aceleração maior. No entanto, desta forma está-se do lado da segurança na medida em que erra por excesso de potência no motor e também não é benéfico ter acelerações bruscas para a longevidade dos componentes.

Note-se que na secção 6-7 calculou-se o binário necessário para segurar a carga como se tratasse da aplicação de binário no sentido do movimento; no entanto, teoricamente, o binário necessário para segurar a carga não teria em conta o rendimento do mecanismo. Este é o valor apresentado no software SIZER da Siemens, mas será utilizado o seguinte valor teórico [1]:

$$T_{6\text{-}7\ \text{-}\ polia} = (200 \times 0 + 200 \times 9,81) \times \frac{0,057}{2} = 55,92\ N.m$$

Consultando [54], encontra-se o modelo de redutores da gama NP que mais se adequa a esta aplicação (modelo NP 035 MF2). Considerando que a velocidade de rotação do servomotor deve ser elevada para que a sua eficiência seja também elevada, procura-se qual a razão de transmissão - $i_{redutora\text{-}aprox}$ que coloca o servomotor na zona de movimento uniforme perto da velocidade nominal média de entrada da caixa redutora. Nessa página do catálogo apresentam-se velocidades médias permitidas de entrada da caixa redutora entre 3100 rpm e 3400 rpm, para 20 ºC; assumindo que a temperatura será maior que 20 ºC, será assumida uma velocidade média de entrada de 3000 rpm.

Procura-se então qual é o $i_{redutora\text{-}aprox}$ que se pode ter, para não exceder a velocidade de rotação do motor de 3000 rpm:

$$i_{redutora\text{-}aprox} = \frac{3000}{\omega_{polia} \times \frac{60}{2\pi}}$$

$$\omega_{polia} = \frac{v_{const}}{\frac{\phi_{polia}}{2}}$$

$$\omega_{polia} = \frac{0,395}{\frac{0,057}{2}} = 13,85\ rad/s$$

$$i_{redutora\text{-}aprox} = \frac{3000}{13,85 \times \frac{60}{2\pi}} = 22,68$$

O valor de razão de transmissão selecionado é $i_{redutora} = 16$ e as características pertinentes da caixa escolhida encontram-se resumidas na tabela A.4. Verifica-se que o binário de aceleração permitido nesta caixa é superior a $T_{0\text{-}1\ \text{-}\ polia}$, e o binário máximo permitido é mais de 1,5 vezes superior a $T_{0\text{-}1}$. O diâmetro do cubo de entrada só pode ser escolhido aquando da seleção do servomotor, mas adianta-se já que se escolherá o de 19 mm.

Estando a caixa redutora selecionada, podem-se calcular os binários $T_{0\text{-}1\ \text{-}\ servo}$ até $T_{6\text{-}7\ \text{-}\ servo}$. Desprezando, numa primeira instância, o binário devido às forças de inércia necessário para acelerar o veio do motor ($J_{servo} = 0\ kg.m^2$, porque ainda não se escolheu um motor), utilizam-se as equações 3.6, 3.7 e 3.8.

[1] A diferença é que no valor teórico não se considera o rendimento,visto que a carga está parada.

Tabela A.4: Características técnicas da caixa redutora escolhida para o eixo B.

Modelo	Binário máx. aceleração (N.m)	Binário máx. permitido (N.m)	$i_{redutora}$	$J_{redutora}$ (kg.cm^2)	$\eta_{redutora}$ (%)
NP 035-MF2	255	$408/1,5 = 272$	16	0,67	90

$$T_{servo\ -\ motor} = (\frac{T_{polia\ -\ motor}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \frac{1}{\eta_{redutora}}$$

$$T_{servo\ -\ gerador} = (\frac{T_{polia\ -\ gerador}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \eta_{redutora}$$

$$T_{0\text{-}1\ -\ servo} = (\frac{98,04}{16} + 0,000067 \times \frac{13,85 \times 16}{0,1}) \times \frac{1}{0,9} = 6,95\ N.m$$

$$T_{1\text{-}2\ -\ servo} = (\frac{69,87}{16} + 0,000067 \times 0) \times \frac{1}{0,9} = 4,85\ N.m$$

$$T_{2\text{-}3\ -\ servo} = (\frac{41,75}{16} + 0,000067 \times \frac{-13,85 \times 16}{0,1}) \times \frac{1}{0,9} = 2,75\ N.m$$

$$T_{3\text{-}4\ -\ servo} = (\frac{26,72}{16} - 0,000067 \times \frac{13,85 \times 16}{0,1}) \times 0,9 = 1,35\ N.m$$

$$T_{4\text{-}5\ -\ servo} = (\frac{44,72}{16} + 0,000067 \times 0) \times 0,9 = 2,52\ N.m$$

$$T_{5\text{-}6\ -\ servo} = (\frac{62,75}{16} + 0,000067 \times \frac{13,65 \times 16}{0,1}) \times 0,9 = 3,66\ N.m$$

$$T_{6\text{-}7\ -\ servo} = (\frac{55,92}{16} + 0,000067 \times 0) \times 1 = 3,50\ N.m$$

O binário efetivo é calculado segundo a equação 3.9:

$$T_{eff} = \left(\frac{0,1 \times 6,95^2 + 1,8 \times 4,85^2 + 0,1 \times 2,75^2 + }{0,1 + 1,8 + 0,1 + } \right.$$
$$\left. \frac{0,1 \times 1,35^2 + 1,8 \times 2,52^2 + 0,1 \times 3,66^2 + 4 \times 3,50^2}{0,1 + 1,8 + 0,1 + 4} \right)^{1/2}$$
$$T_{eff} = 3,7\ N.m$$

Neste caso, entendeu-se ainda que no caso de o servomotor ficar parado durante um grande período de tempo ($T_{eff} = 3,5\ N.m$), não se deve exceder 75 % do seu binário de arranque em regime permanente. Em termos numéricos, binário de arranque em regime permanente $>= \frac{3,5}{0,75} >= 4,67\ N.m$.

Começando pela marca Siemens, o servomotor escolhido foi o 1FK7061-4CF7, cujas características podem ser consultadas na tabela A.5. O binário médio efetivo está a 71 % do limite recomendado, o binário máximo está a 40 % do limite, o binário de arranque está também abaixo do limite considerado e a velocidade nominal não chega a ser atingida.

A razão de inércias é dada pelas equações 3.10 e 3.11:

Tabela A.5: Características técnicas do servomotor Siemens escolhido para o eixo B.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade nominal (rpm)	M_0 (N.m)	J_{servo} (kg.cm^2)
1FK7061-4CF7	17,3	5,40	3000	6,40	5,10

$$\frac{1}{2} \times J_{eq} \times {\omega_{servo}}^2 = \frac{1}{2} \times 200 \times \left(\frac{\omega_{servo} \times \frac{0,057}{2}}{16} \right)^2 + \frac{1}{2} \times 0,000067 \times {\omega_{servo}}^2$$

$$J_{eq} = 7,02 \ kg.cm^2$$
$$r_{inércias} = \frac{7,02}{5,10} = 1,38$$

Fica assim selecionado o motor da marca Siemens para o eixo B.

Do lado da marca RA, o servomotor escolhido é o MPL-B430P, cujas características podem ser consultadas na tabela A.6. O binário médio efetivo está a 74 % do limite recomendado, o binário máximo está a 69 % do limite, o binário de arranque também está abaixo do limite considerado e não se está a atingir a velocidade máxima do servomotor.

Tabela A.6: Características técnicas do servomotor Rockwell Automation escolhido para o eixo B.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade máxima (rpm)	M_0 (N.m)	J_{servo} (kg.cm^2)
MPL-B430P	19,80	4,20	5000	6,55	4,20

A razão de inércias é dada pelas equações 3.10 e 3.11:

$$J_{eq} = 7,02 kg.cm^2$$
$$r_{inércias} = \frac{7,02}{4,2} = 1,67$$

Poder-se-ia arguir que a elevada inércia do servomotor escolhido teria um efeito significativo nos valores de binário necessário previamente calculados. Isso é verdade, relativamente ao binário máximo, mas o binário efetivo ficaria sensivelmente igual, visto que não tem influência no ciclo todo e até é benéfico na secção 2-3. Como verificação, considera-se um motor com $J_{servo} = 5,10 \ kg.cm^2$ e calcula-se novamente um valor de binário máximo e o binário efetivo:

$$T_{0\text{-}1 \ \text{-} \ servo} = 8,23 \ N.m$$
$$T_{eff} = 3,74 \ N.m$$

Pelo que não há problemas levantados devido à assunção inicial de que se podia desprezar o momento de inércia do servomotor. Fica assim selecionado o motor da RA para o eixo B, pelo que se dá como concluída a seleção de servomotores para este eixo.

Eixos C

Estes eixos correspondem aos movimentos horizontais do LASER e da câmara, que são acionados independentemente por dois conjuntos servomotor - caixa redutora - polia - correia independentes. Como são duas aplicações tão semelhantes, foi um requisito por parte da empresa que se utilizassem servomotores iguais. Novamente, chama-se a atenção do leitor para a figura 2.4, na qual estes eixos estão bem visíveis.

A utilização de servomotores nesta aplicação justifica-se com a necessidade de posicionar com precisão estes equipamentos para garantir a qualidade durante a gravação. Visto que se trata de cargas leves e pouco dinâmicas, poder-se-ia recorrer alternativamente a motores passo-a-passo. No entanto, optou-se por servomotores com o intuito de uniformizar o tipo de motores utilizados.

O ciclo considerado consiste num avanço e um recuo com duração de 2 s cada, resultando num tempo total de trabalho de 4 s. Ao considerar DC = 0,5, obtém-se um tempo total de ciclo de 8 s.

Será assumido que o eixo percorre toda a distância do seu curso (600 mm) durante o movimento e um tempo de aceleração e desaceleração de 100 ms. A massa da carga considerada é 50 kg.

Sabe-se que a polia tem um diâmetro de 41 mm e assume-se um rendimento do mecanismo correia-polia (incluindo guias) de 80 %. As curvas de posição e velocidade para o movimento durante o ciclo, estão apresentadas na figura A.6.

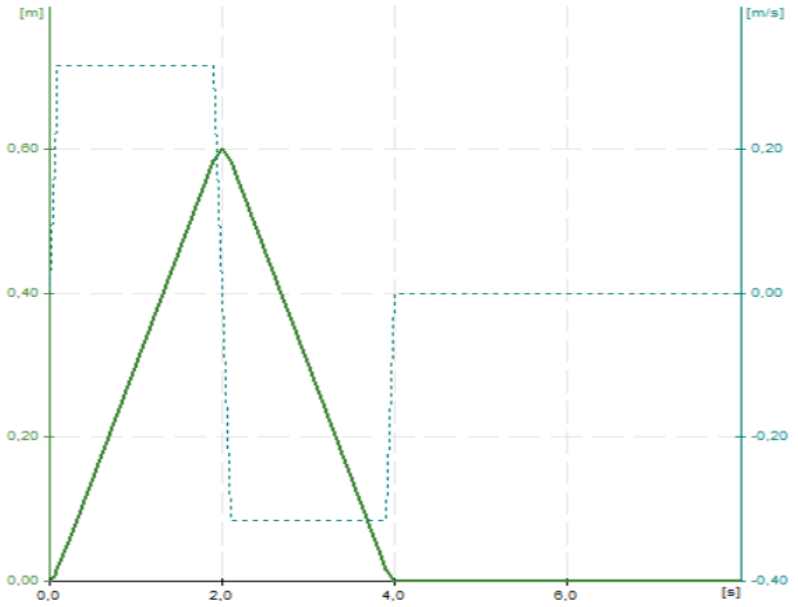


Figura A.6: Perfil temporal da posição - x_{carga} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo C. Fonte: SIZER da Siemens.

Com os valores conhecidos substituídos, pode-se calcular a distância percorrida

através da equação 3.1:

$$0,6 = \int_{0}^{0,1} (\frac{v_{const}}{0,1} \times t)\ dt + v_{const} \times (1,8 - 0,1 - 0,1) + \int_{2-0,1}^{1} (v_{const} - \frac{v_{const}}{0,1} \times t)\ dt$$

Integrando:

$$0,6 = v_{const} \times 0.05 + v_{const} \times 1.8 + v_{const} \times 0.05$$

Resolvendo em ordem a v_{const}:

$$v_{const} = 0.316\ m/s$$

A aceleração - $a_{inicial}$, que é igual à desaceleração, é calculada usando a equação 3.2:

$$a_{inicial} = \frac{0.316}{0.1} = 3,16\ m/s^2$$

O perfil de binário - T_{polia} e velocidade angular em rpm - n_{polia} na polia durante o ciclo considerado está representado na figura A.7.

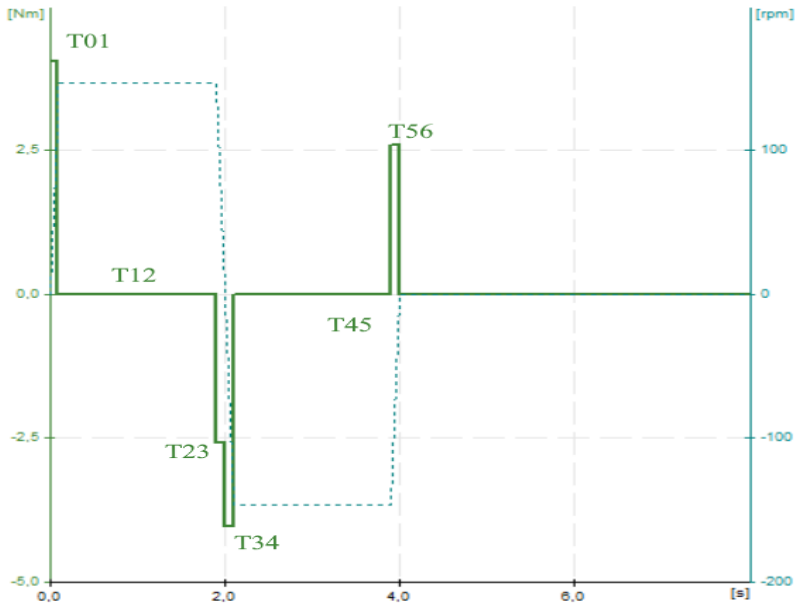


Figura A.7: Perfil temporal do binário na polia - T_{polia} (traço contínuo) e velocidade angular da polia - ω_{polia} (traço interrompido) do eixo acionado pelo servomotor C. Fonte: SIZER da Siemens.

Note-se que ao longo do ciclo, a variável T_{polia} pode tomar seis valores distintos, cujo significado é apresentado abaixo:

- $T_{0\text{-}1}$ corresponde ao arranque no avanço;
- $T_{1\text{-}2}$ corresponde ao avanço em movimento uniforme;

- $T_{2\text{-}3}$ corresponde à frenagem no avanço;
- $T_{3\text{-}4}$ corresponde ao arranque no recuo;
- $T_{4\text{-}5}$ corresponde ao recuo em movimento uniforme;
- $T_{5\text{-}6}$ corresponde à frenagem no recuo.

Segue-se o cálculo destes valores através das equações 3.4 e 3.3:

$$T_{polia\ -\ motor} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \frac{1}{\eta_{c\text{-}p}} \times \frac{\phi_{polia}}{2}$$

$$T_{polia\ -\ gerador} = (m_{carga} \times a_{cel\text{-}carga} + F_{atrito}) \times \eta_{c\text{-}p} \times \frac{\phi_{polia}}{2}$$

$$T_{0\text{-}1\ -\ polia} = (50 \times 3,16 + 0) \times \frac{1}{0,8} \times \frac{0,041}{2} = 4,05\ N.m$$

$$T_{1\text{-}2\ -\ polia} = (50 \times 0 + 0) \times \frac{1}{0,8} \times \frac{0,041}{2} = 0\ N.m$$

$$T_{2\text{-}3\ -\ polia} = (50 \times (-3,16) + 0) \times 0,8 \times \frac{0,041}{2} = -2,59\ N.m$$

$$T_{3\text{-}4\ -\ polia} = (50 \times (-3,16) + 0) \times \frac{1}{0,8} \times \frac{0,041}{2} = -4,05\ N.m$$

$$T_{4\text{-}5\ -\ polia} = (50 \times 0 + 0) \times \frac{1}{0,8} \times \frac{0,041}{2} = 0,00\ N.m$$

$$T_{5\text{-}6\ -\ polia} = (50 \times 3,16 + 0) \times 0,8 \times \frac{0,041}{2} = 2,59\ N.m$$

Consultando [54], encontra-se o modelo de redutores da gama NP que mais se adequa a esta aplicação (modelo NP 005 MF2). Considerando que a velocidade de rotação do servomotor deve ser elevada para que a sua eficiência seja também elevada, procura-se qual a razão de transmissão - $i_{redutora\text{-}aprox}$ que coloca o servomotor na zona de movimento uniforme perto da velocidade nominal média de entrada da caixa redutora. Nessa página do catálogo apresentam-se velocidades médias permitidas de entrada da caixa redutora de 4000 rpm, para 20 ºC.

Assumindo que a temperatura será maior que 20 ºC, será assumida uma velocidade média de entrada de 3500 rpm. Procura-se então qual é o $i_{redutora\text{-}aprox}$ que se pode ter, para não exceder a velocidade de rotação de entrada de 3500 rpm:

$$i_{redutora\text{-}aprox} = \frac{3500}{\omega_{polia} \times \frac{60}{2\pi}}$$

$$\omega_{polia} = \frac{v_{const}}{\frac{\phi_{polia}}{2}}$$

$$\omega_{polia} = \frac{0,316}{\frac{0,041}{2}} = 15,41\ rad/s$$

$$i_{redutora\text{-}aprox} = \frac{3500}{15,41 \times \frac{60}{2\pi}} = 23,78$$

O valor de razão de transmissão selecionado é $i_{redutora} = 20$ e as características pertinentes da caixa escolhida encontram-se resumidas na tabela A.7. Verifica-se que o binário de aceleração permitido nesta caixa é superior a $T_{0\text{-}1}$ e o binário máximo permitido é mais de 1,5 vezes superior a $T_{0\text{-}1}$. O diâmetro do cubo de entrada só pode ser escolhido aquando da seleção do servomotor, mas adianta-se já que se escolherá o de 8 mm para servomotor Siemens e 9 mm para o da RA.

Tabela A.7: Características técnicas da caixa redutora escolhida para o eixo A.

Modelo	Binário máx. aceleração (N.m)	Binário máx. permitido (N.m)	$i_{redutora}$	$J_{redutora}$ (kg.cm^2)	$\eta_{redutora}$ (%)
NP 005s-MF2	11	$18/1,5 = 12$	20	0,03	90

Estando a caixa redutora selecionada, podem-se calcular os binários $T_{0\text{-}1\ \text{-}\ servo}$ até $T_{5\text{-}6\ \text{-}\ servo}$. Desprezando, numa primeira instância, o binário devido às forças de inércia necessário para acelerar o veio do motor ($J_{servo} = 0\ kg.m^2$, porque ainda não se escolheu um motor), utilizam-se as equações 3.6, 3.7 e 3.8.

$$T_{servo\ \text{-}\ motor} = (\frac{T_{polia\ \text{-}\ motor}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \frac{1}{\eta_{redutora}}$$

$$T_{servo\ \text{-}\ gerador} = (\frac{T_{polia\ \text{-}\ gerador}}{i_{redutora}} + J_{redutora} \times \frac{\omega_{polia} \times i_{redutora}}{t_{acel}}) \times \eta_{redutora}$$

$$T_{0\text{-}1\ \text{-}\ servo} = (\frac{4,05}{20} + 0,000003 \times \frac{15,41 \times 20}{0,1}) \times \frac{1}{0,9} = 0,24\ N.m$$

$$T_{1\text{-}2\ \text{-}\ servo} = (\frac{0}{20} + 0,000003 \times 0) \times \frac{1}{0,9} = 0\ N.m$$

$$T_{2\text{-}3\ \text{-}\ servo} = (\frac{-2,59}{20} + 0,000003 \times \frac{-15,41 \times 20}{0,1}) \times 0,9 = -0.13\ N.m$$

$$T_{3\text{-}4\ \text{-}\ servo} = (\frac{-4,05}{20} + 0,000003 \times \frac{-15,41 \times 20}{0,1}) \times \frac{1}{0,9} = -0,24\ N.m$$

$$T_{4\text{-}5\ \text{-}\ servo} = (\frac{0}{20} + 0,000003 \times 0) \times \frac{1}{0,9} = 0,00\ N.m$$

$$T_{5\text{-}6\ \text{-}\ servo} = (\frac{2,59}{20} + 0,000003 \times \frac{15,41 \times 20}{0,1}) \times 0,9 = 0,13\ N.m$$

O binário efetivo é calculado segundo a equação 3.9:

$$T_{eff} = \left(\frac{0,1 \times 0,24^2 + 1,8 \times 0 + 0,1 \times 0.13^2 + }{0,1 + 1,8 + 0,1 + } \right.$$
$$\left. \frac{0,1 \times 0,24^2 + 1,8 \times 0 + 0,1 \times 0,13^2 + 2 \times 0}{0,1 + 1,8 + 0,1 + 4} \right)^{1/2}$$
$$T_{eff} = 0,04\ N.m$$

Começando pela marca Siemens, o servomotor escolhido foi o 1FK7011-5AK7, cujas características podem ser consultadas na tabela A.8. O binário médio efetivo está a

30 % do limite recomendado, o binário máximo está a 51 % do limite e não se está a atingir a velocidade nominal do servomotor.

Tabela A.8: Características técnicas do servomotor Siemens escolhido para o eixo A.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade nominal (rpm)	J_{servo} (kg.cm^2)
1FK7022-5AK7	0,50	0,08	6000	0,08

A razão de inércias é dada pelas equações 3.10 e 3.11:

$$\frac{1}{2} \times J_{eq} \times {\omega_{servo}}^2 = \frac{1}{2} \times 50 \times \left(\frac{\omega_{servo} \times \frac{0{,}041}{2}}{20} \right)^2 + \frac{1}{2} \times 0,000003 \times {\omega_{servo}}^2$$

$$J_{eq} = 0,556\ kg.cm^2$$
$$r_{inércias} = \frac{0,556}{0,08} = 6,95$$

Do lado da marca RA, o servomotor escolhido é o MPL-B1510U, cujas características podem ser consultadas na tabela A.9. O binário médio efetivo está a 32 % do limite recomendado, o binário máximo está a 36 % do limite e não se está a atingir a velocidade máxima do servomotor.

Tabela A.9: Características técnicas do servomotor Rockwell Automation escolhido para o eixo A.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade máxima (rpm)	J_{servo} (kg.cm^2)
MPL-B1530U	0,77	0,19	7000	0,074

A razão de inércias é dada pelas equações 3.10 e 3.11:

$$J_{eq} = 0,556\ kg.cm^2$$
$$r_{inércias} = \frac{0,556}{0,074} = 7,5$$

Neste caso, verifica-se que, apesar de o servomotor estar folgado, a sua razão de inércias está ligeiramente alta tanto para o servomotor Siemens como estará para o servomotor da RA. Este valor não é crítico e será aceite por ser uma aplicação pouco dinâmica, mas a seleção dos motores imediatamente acima seria de considerar caso não houvesse diferença significativa em termos de preço. Ficam assim selecionados os servomotores para estes eixos.

É importante acrescentar uma nota relativamente a um leitor que possa estar a seguir a leitura deste capítulo com o projeto feito no software Motion Analyzer da RA. Ao selecionar o servomotor MPL-B1510U com um módulo de motor da série Kinetix 5700 (que é o caso desta aplicaçao), o software levanta um aviso relativamente à razão entre a corrente nominal do módulo (3,5 A) e a corrente correspondente ao binário de arranque do servomotor (0,95 A). Isto deve-se ao facto de a razão entre estes valores ser maior que 2: $3,5 \div 0,95 = 3,7$.

Este aviso significa que o módulo ou o servomotor estão mal dimensionados um para o outro e pode haver problemas de controlo do servomotor. O problema é: trata-se do módulo de menor capacidade desta série e, portanto, não há nenhuma outra opção. A solução para isto encontra-se no manual da RA denominado Kinetix 5700 Drive System Design Guide, referenciado em [55], no capítulo Kinetix 5700 (400V operation) Drives/Kinetix MPL Servo Motor Curves, onde está explicitado que este motor é compatível com este módulo.

Eixo D

O último eixo a analisar é diferente dos anteriores na medida em que constitui uma aplicação angular, enquanto os restantes eram lineares. Neste caso, trata-se de um servomotor acoplado a uma caixa redutora com um sistema de engrenagens cónicas que permite saída numa direção a 90º da entrada. Adicionalmente, o seu cubo de saída é oco para que a ele seja acoplada uma mesa rotativa através de uma chaveta (figura A.8).

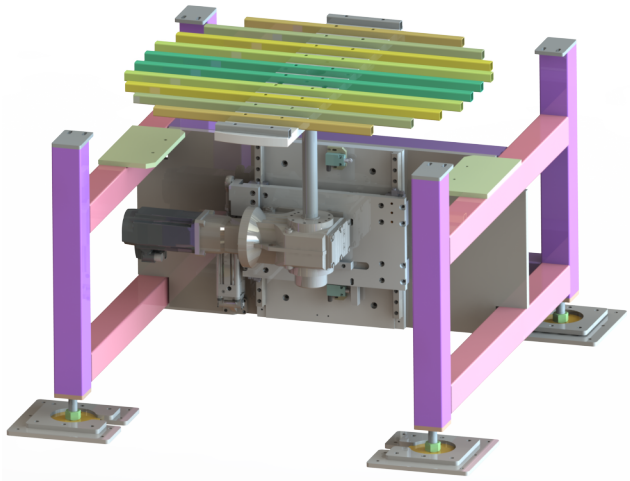


Figura A.8: Componentes que constituem o eixo D. Projeto: Gislotica.

A utilização de servomotores nesta aplicação justifica-se com a necessidade de posicionar com precisão o pneu angularmente e se tratar de uma carga com alguma inércia.

O ciclo considerado consiste numa rotação com duração de 1 s com DC = 0,5, resultando num tempo total de ciclo de 2 s. Será assumido que o eixo dá uma volta completa (360º) durante o movimento e um tempo de aceleração e desaceleração de 200 ms. As curvas de aceleração e velocidade angulares estão representadas na figura A.9.

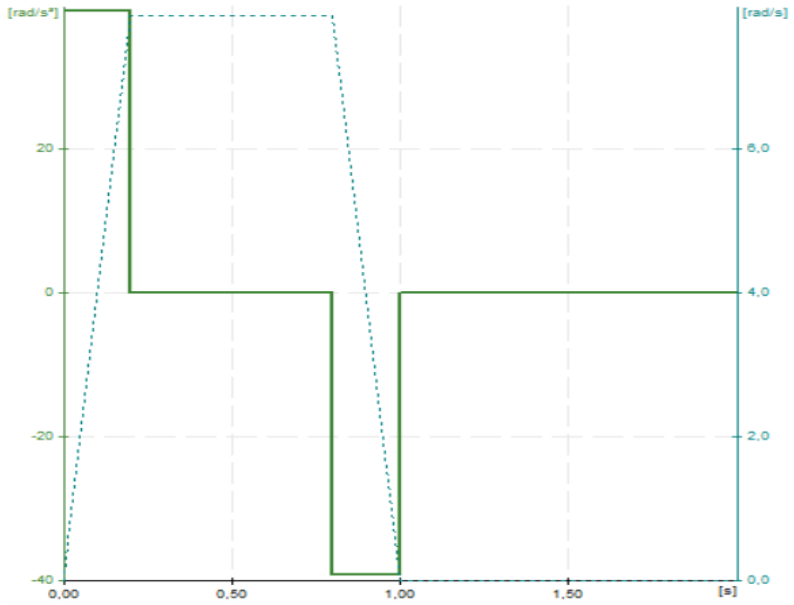


Figura A.9: Perfil temporal da aceleração - a_{cel} (traço contínuo) e velocidade - v_{carga} (traço interrompido) da carga do eixo D. Fonte: SIZER da Siemens.

Sabe-se que a carga[2] tem uma inércia de $3,3\ kg.m^2$, calculada no *Solidworks* e será assumido um rendimento deste componente de 90 %. A caixa redutora é da marca SEW e já havia sido previamente escolhida pela empresa. As suas características pertinentes encontram-se resumidas na tabela A.10, e apenas tem disponíveis acoplamentos para veios de entrada com diâmetro 19 mm, pelo que se deverá ter isto em conta aquando da seleção de um servomotor.

Tabela A.10: Características técnicas da caixa redutora escolhida para o eixo D.

Modelo	Binário máx. permitido (N.m)	$i_{redutora}$	$J_{redutora}$ (kg.cm^2)	$\eta_{redutora}$ (%)
KH47 /R AQA 100/3	$400/1,5 = 270$	48,95	0,80	90

Segue-se a explicação do código do modelo desta caixa redutora:

- KH Redutor cónico com veio oco;
- 47 Relativo à dimensão das engrenagens;
- $/R$ Efeito de *backlash* reduzido;
- AQA Aptador à entrada para servomotor com veio enchavetado;
- 100/3 Dimensão do adaptador: flange de lado 100 mm, variante 3;
- $1400/29rpm$ Razão de transmissão.

Com os valores conhecidos substituídos, pode-se calcular a distância percorrida através da equação 3.1:

$$2\pi = \int_{0}^{0,2} (\frac{\omega_{const}}{0,2} \times t)\ dt + \omega_{const} \times (1 - 0,2 - 0,2) + \int_{0,8}^{1-0,2} (\omega_{const} - \frac{\omega_{const}}{0,2} \times t)\ dt$$

[2]A carga inclui os componentes que constituem a mesa e um pneu do tipo PCR, fornecido pela empresa.

Integrando:

$$2\pi = \omega_{const} \times 0.1 + \omega_{const} \times 0,6 + \omega_{const} \times 0.1$$

Resolvendo em ordem a ω_{const}:

$$\omega_{const} = 7,854\ rad/s$$

A aceleração - $\dot{\omega}_{inicial}$, que é igual à desaceleração, é calculada usando a equação 3.2:

$$\dot{\omega}_{inicial} = \frac{7,854}{0.2} = 39,27\ rad/s^2$$

O binário máximo - T_{max} aplicado na carga dá-se no instante inicial e é importante para verificar se a caixa redutora está dimensionada para tal.

$$T_{max} = \dot{\omega}_{inicial} \times J_{carga} = 39,27 \times 3,3 = 130\ N.m$$

Comparando este resultado com o binário máximo permitido da tabela A.10, verifica-se que o máximo binário na carga (130 N.m) é bastante inferior ao binário permitido de regime permanente (400 N.m), pelo que a caixa redutora está apta para esta aplicação.

O perfil de binário - T_{motor} e velocidade angular em rpm - n_{motor} no motor durante o ciclo considerado está representado na figura A.10.

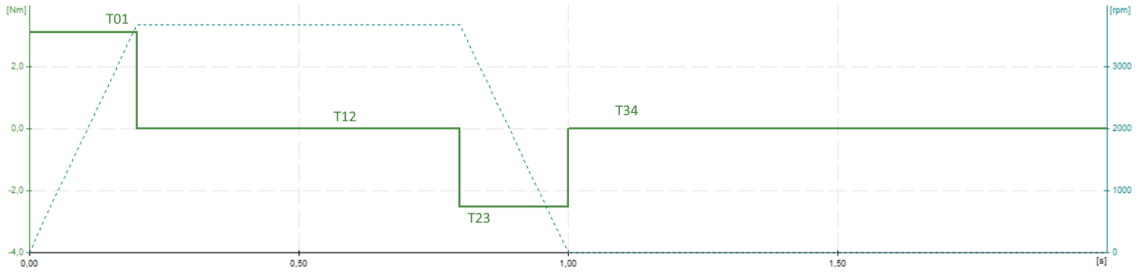


Figura A.10: Perfil temporal do binário - T_{motor} (traço contínuo) e velocidade angular - ω_{servo} (traço interrompido) do servomotor D. Fonte: SIZER da Siemens.

Note-se que ao longo do ciclo, a variável T_{motor} pode tomar quatro valores distintos, cujo significado é apresentado abaixo:

- $T_{0\text{-}1}$ corresponde ao arranque no avanço;
- $T_{1\text{-}2}$ corresponde ao avanço em movimento uniforme;
- $T_{2\text{-}3}$ corresponde à frenagem no avanço;
- $T_{3\text{-}4}$ eixo parado.

Segue-se o cálculo destes valores através das equações 3.4 e 3.3:

$$T_{servo\ -\ motor} = (\frac{J_{carga}}{i_{redutora}} + J_{redutora} \times i_{redutora}) \times a_{cel\text{-}carga} \times \frac{1}{\eta_{redutora}}$$

$$T_{servo\ -\ gerador} = (\frac{J_{carga}}{i_{redutora}} + J_{redutora} \times i_{redutora}) \times a_{cel\text{-}carga} \times \eta_{redutora}$$

$$T_{0\text{-}1\ -\ servo} = (\frac{3,3}{48,95} + 0,00008 \times 48,95) \times 39,27 \times \frac{1}{0,9} = 3,11\ N.m$$

$$T_{1\text{-}2\ -\ servo} = (\frac{3,3}{48,95} + 0,00008 \times 48,95) \times 0 \times \frac{1}{0,9} = 0\ N.m$$

$$T_{2\text{-}3\ -\ servo} = (\frac{3,3}{48,95} + 0,00008 \times 48,95) \times -39,27 \times 0,9 = -2,52\ N.m$$

$$T_{3\text{-}4\ -\ servo} = (\frac{3,3}{48,95} + 0,00008 \times 38,95) \times 0 \times \frac{1}{0,9} = 0\ N.m$$

O binário efetivo é calculado segundo a equação 3.9:

$$T_{eff} = \left(\frac{0,2 \times 3,11^2 + 0,6 \times 0 + 0,2 \times 2,52^2}{0,2 + 0,6 + 0,2 + 1}\right)^{1/2}$$

$$T_{eff} = 1,27\ N.m$$

Começando pela marca Siemens, o servomotor escolhido foi o 1FK7042-2AF7, cujas características podem ser consultadas na tabela A.11. O binário médio efetivo está a 56 % do limite recomendado, o binário máximo está a 68 % do limite e atinge-se uma velocidade acima da velocidade nominal do motor, pelo que é preciso ver este último aspeto em detalhe.

Tabela A.11: Características técnicas do servomotor Siemens escolhido para o eixo D.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade nominal (rpm)	J_{servo} (kg.cm^2)
1FK7042-2AF7	10,50	2,60	3000	3,20

A curva de funcionamento deste motor está apresentada na figura A.11, onde se podem ver as curvas limite para a controlo com limite de tensão e controlo com enfraquecimento de campo. A Siemens especifica na secção *Torque limit when operating on a SINAMICS S110 / S120 with field weakening* em [56] que: utilizando uma unidade de controlo da gama SINAMICS S110/120, é possível efetuar enfraquecimento de campo para conseguir atingir velocidades mais elevadas.

O enfraquecimento de campo é o nome dado a uma técnica utilizada no controlo vetorial de servomotores. Normalmente, a tensão induzida pelos ímanes permanentes nos enrolamentos do estator (também chamada de força contra eletromotriz) aumenta à medida que a velocidade aumenta e, portanto, cada vez é menor a tensão resultante para gerar corrente no motor. Através de controlo vetorial, é possível

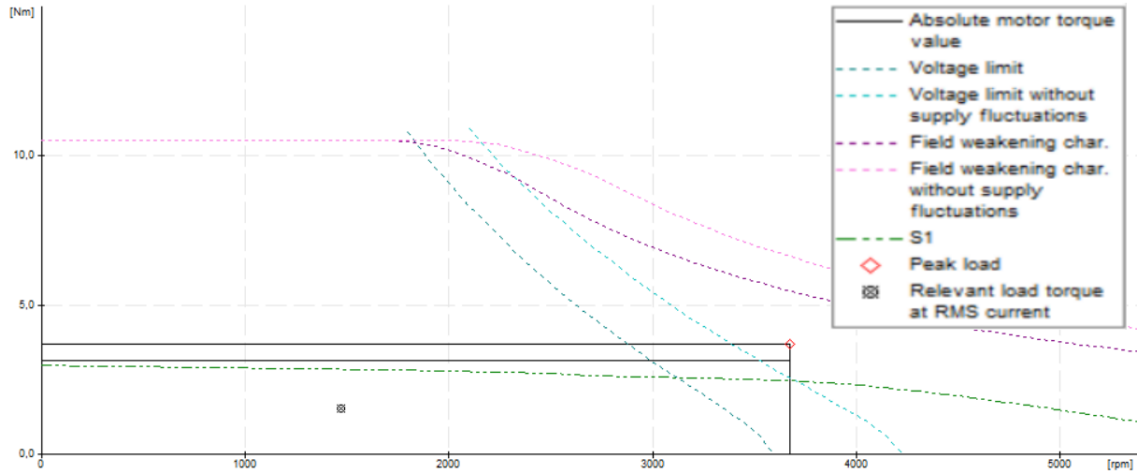


Figura A.11: Curva de funcionamento do servomotor D. Fonte: SIZER da Siemens.

comandar a corrente geradora de binário (chamada de corrente de quadratura) e a corrente geradora de campo magnético nos enrolamentos (chamada de corrente direta), através da variação do atraso de fase entre a corrente de alimentação e tensão induzida pelos ímanes permanentes, conhecida através da posição do servomotor.

O objetivo é ajustar a corrente direta de tal modo que o fluxo magnético resultante nos enrolamentos seja diminuído (daí enfraquecimento de campo), diminuindo a tensão induzida nos enrolamentos do estator. Desse modo, está disponível maior tensão de alimentação para contribuir para o aumento de velocidade do servomotor.

Visto que nesta aplicação se irá utilizar uma unidade de controlo dessa gama, então pode-se escolher este servomotor.

A razão de inércias para o servomotor Siemens é dada pelas equações 3.10 e 3.11:

$$\frac{1}{2} \times J_{eq} \times {\omega_{servo}}^2 = \frac{1}{2} \times 3,3 \times \left(\frac{\omega_{servo}}{48,95}\right)^2 + \frac{1}{2} \times 0,00008 \times {\omega_{servo}}^2$$

$$J_{eq} = 14,5\ kg.cm^2$$
$$r_{inércias} = \frac{14,5}{3,2} = 4,55$$

Do lado da marca RA, o servomotor escolhido é o MPL-B420P, cujas características podem ser consultadas na tabela A.12. O binário médio efetivo está a 38 % do limite recomendado, o binário máximo está a 62 % do limite e não é atingida a velocidade máxima do servomotor.

A razão de inércias é dada pelas equações 3.10 e 3.11:

$$J_{eq} = 14,5\ kg.cm^2$$
$$r_{inércias} = \frac{14,5}{3,0} = 4,83$$

Tabela A.12: Características técnicas do servomotor Rockwell Automation escolhido para o eixo A.

Modelo	Binário máx. (N.m)	Binário nominal (N.m)	Velocidade máxima (rpm)	J_{servo} (kg.cm^2)
MPL-B1530U	13,5	3,69	75000	3,00

Neste caso, idealmente, ter-se-ia escolhido o servomotor de modelo imediatamente inferior (MPL-B330P) porque as suas características também se adequariam. Contudo, esse servomotor tem um veio de 16 mm, incompatível[3] com o diâmetro de entrada do cubo da redutora de 19 mm.

Fica assim concluída a seleção de servomotores para este eixo, e para a máquina.

[3]Poder-se-ia utilizar uma manga no veio do servomotor que compensasse o seu diâmetro menor que o do cubo da caixa redutora, mas essa estratégia não foi aqui considerada.

Referências bibliográficas do anexo A

[54] Wittenstein Alpha. alpha basic line & alpha value line product catalog, 2020.

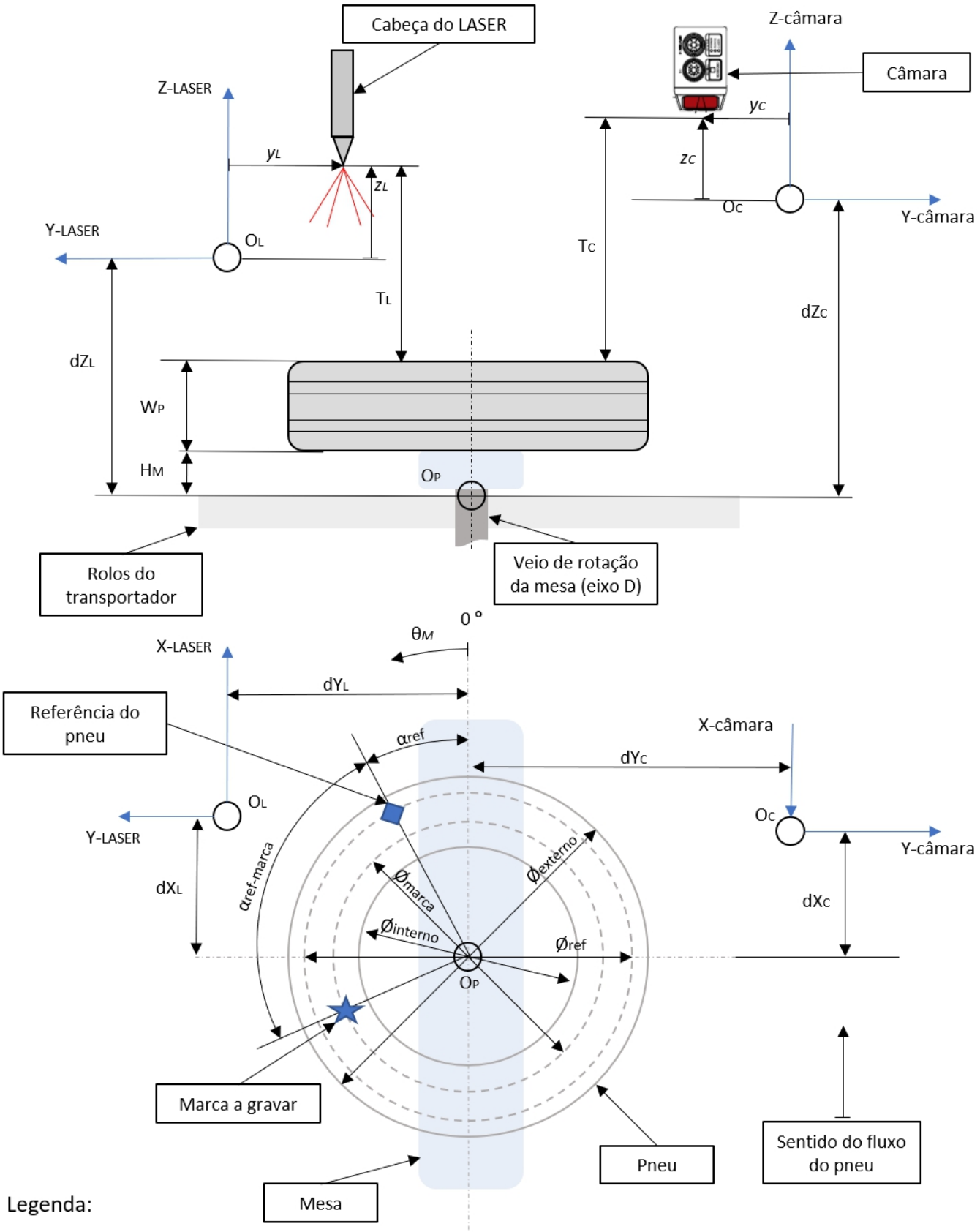
[55] Rockwell Automation. Kinetix 5700 drive system design guide, 2020. URL https://literature.rockwellautomation.com/idc/groups/literature/documents/rm/knx-rm010_-en-p.pdf.

[56] Siemens. Simotics - drive technology - 1fk7 g2 synchronous motors, 2018. URL https://support.industry.siemens.com/cs/document/55379345/simotics-s-1fk7-g2-synchronous-motors?dti=0&dl=en&pnid=13349&lc=pt-BR.

Todas as referências com página web se encontravam disponíveis a 28 de junho de 2021, data de entrega da versão provisória da dissertação.

Anexo B

Desenho esquemático da posição do pneu durante o processo

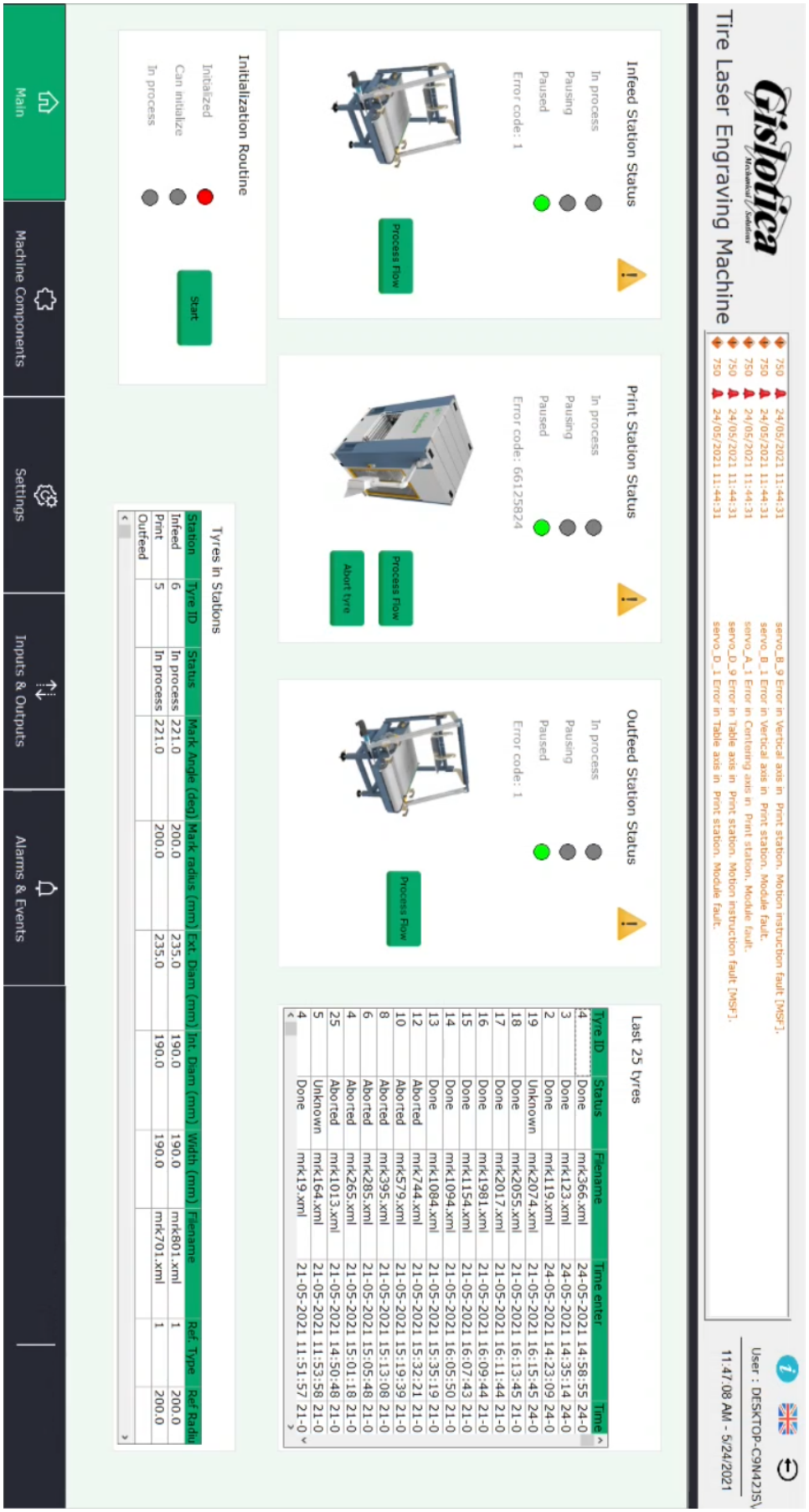


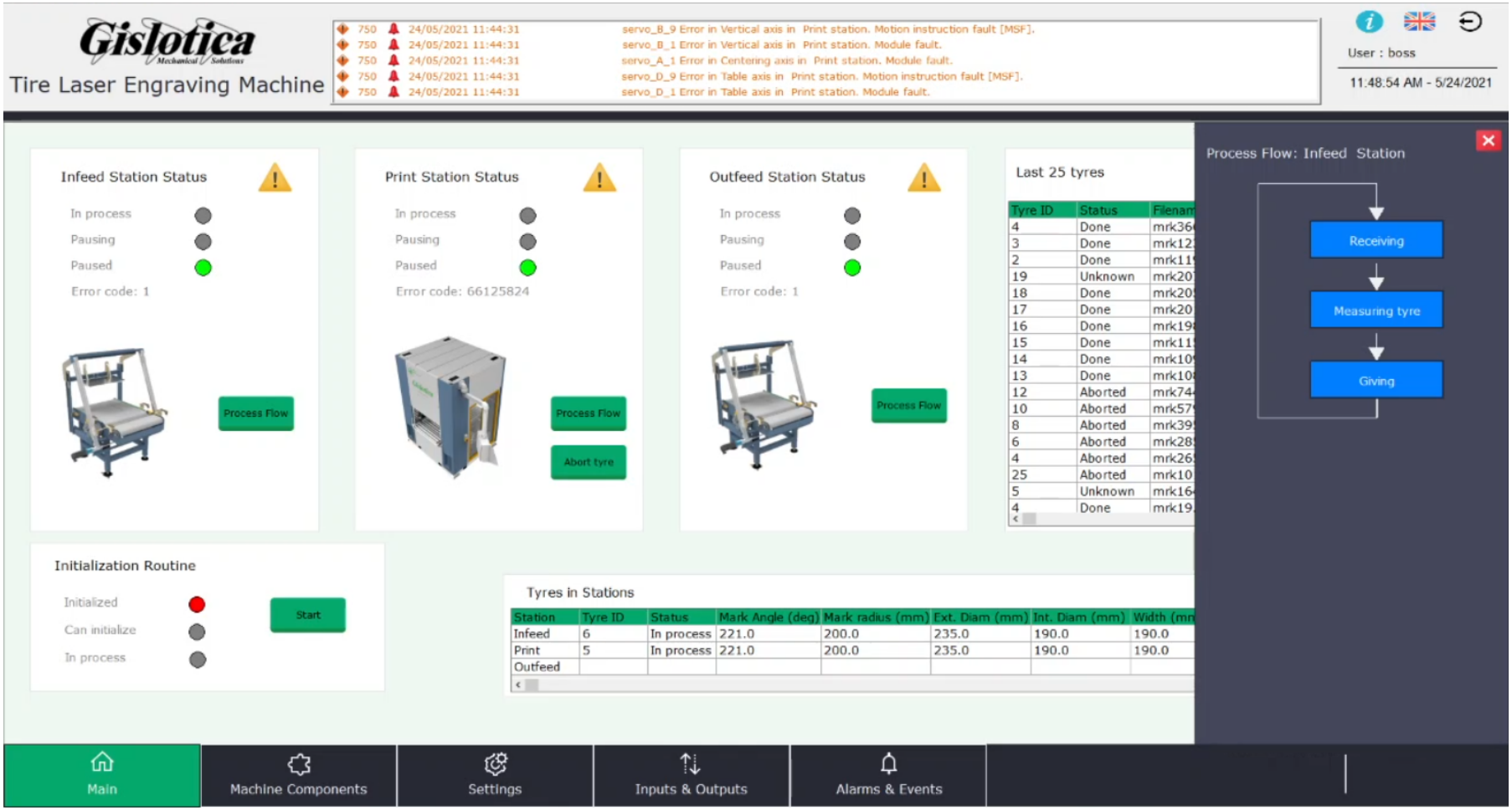
Legenda:

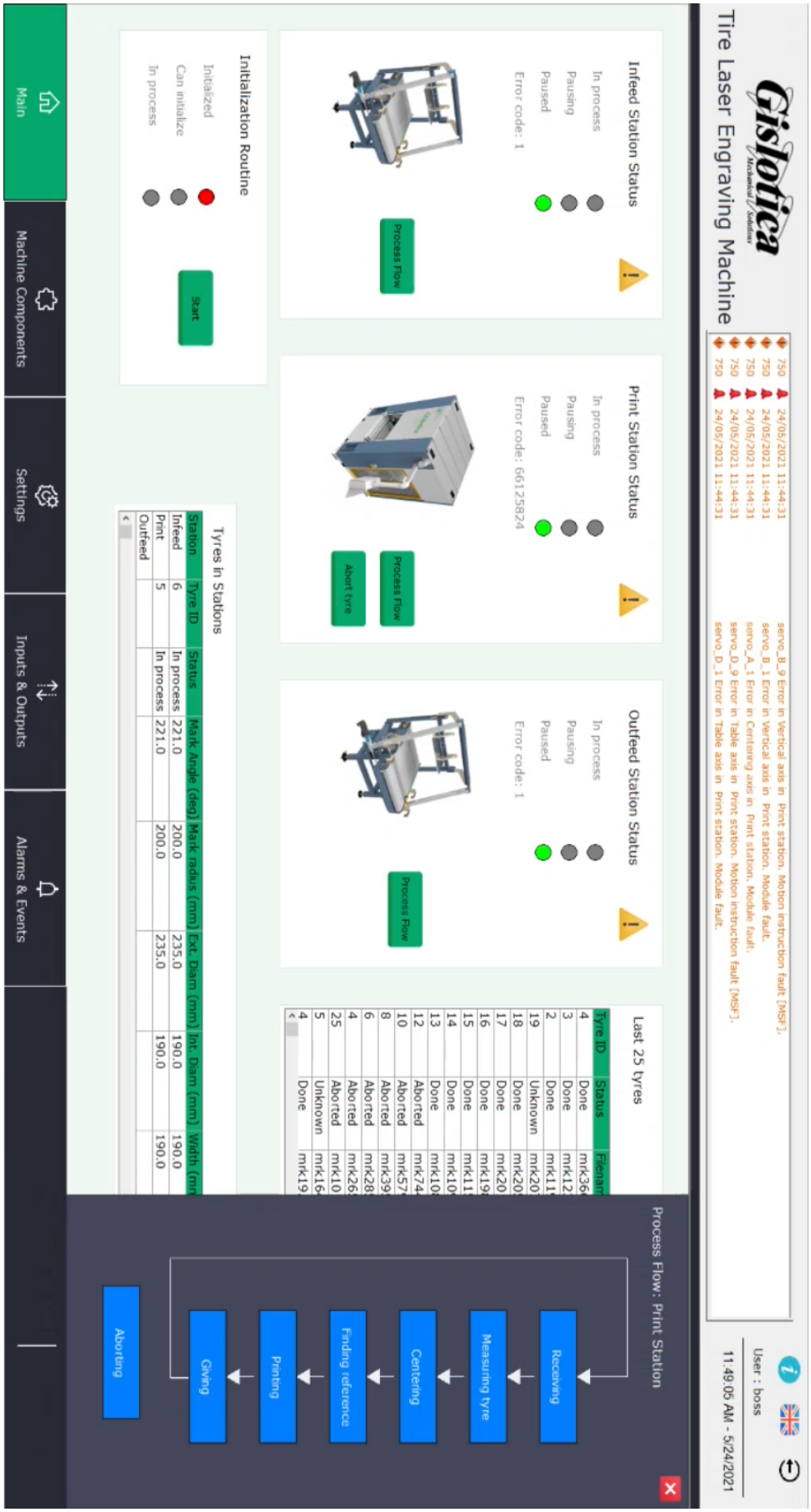
- Eixos Z: Eixo na direção vertical, com o sentido de baixo para cima;
- Eixos Y: Eixo na direção perpendicular ao do fluxo do pneu;
- Eixos X: Eixo na direção do fluxo do pneu;
- z_L, z_C, y_L e y_C : variáveis de posição segundo as direções dos eixos Z e Y do LASER e da câmara;
- dX_L, dY_L, dZ_L , dX_C, dY_C, dZ_C: Distância segundo os eixos X, Y e Z entre o ponto O_P e os pontos O_L e O_C, medidas diretamente na máquina;
- Pontos O_L e O_C: Pontos tomados pela lente do LASER e o emissor da câmara quando os eixos B e C (respetivos) estão na posição Inicial (Home);
- Ponto O_P: Origem do referencial do pneu sob o transportador;
- θ_M: Variável de posição angular da mesa.
- α_{ref}: Ângulo entre a referência e a posição da mesa;
- $\alpha_{ref\text{-}marca}$: Ângulo entre a referência e o ponto de marcação;
- Ø: Diâmetros pertinentes para a marcação;
- W_P: Largura do pneu;
- H_M: Distância entre a superfície da mesa e a superfície dos rolos;
- T_L e T_C: Distância de trabalho do LASER e da câmara.

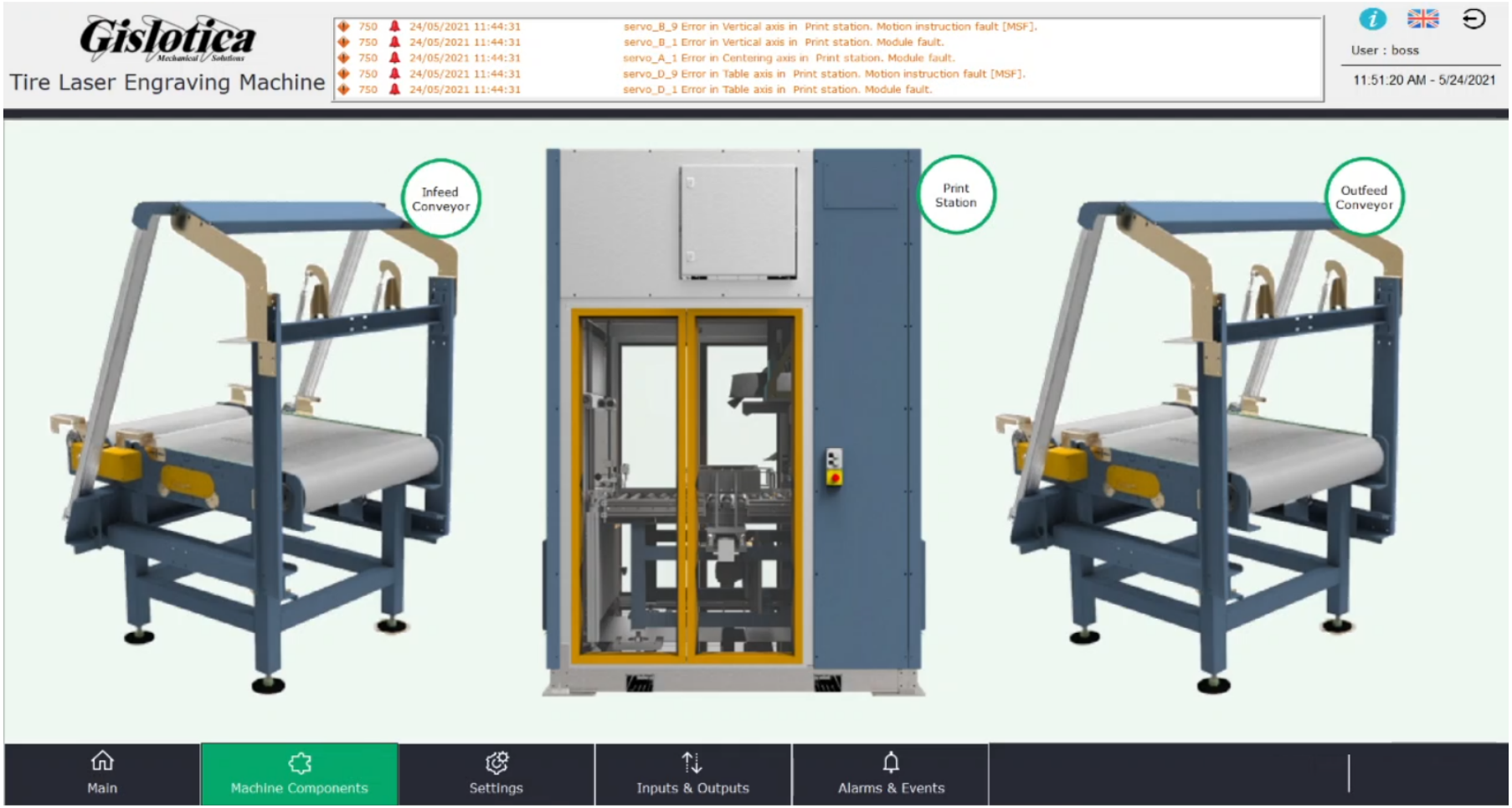
Anexo C

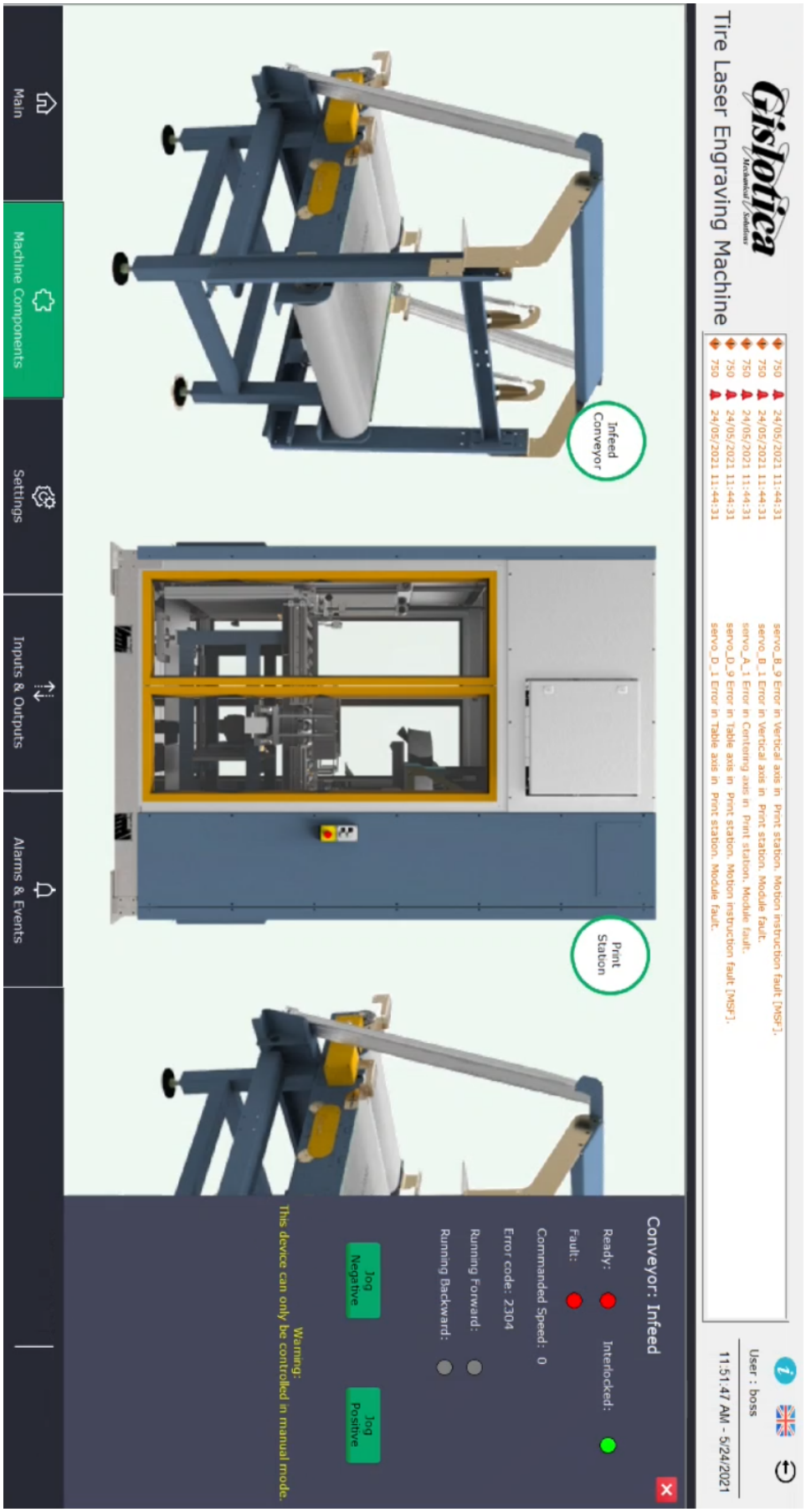
Ecrãs criados no FactoryTalk SE para interface com o operador

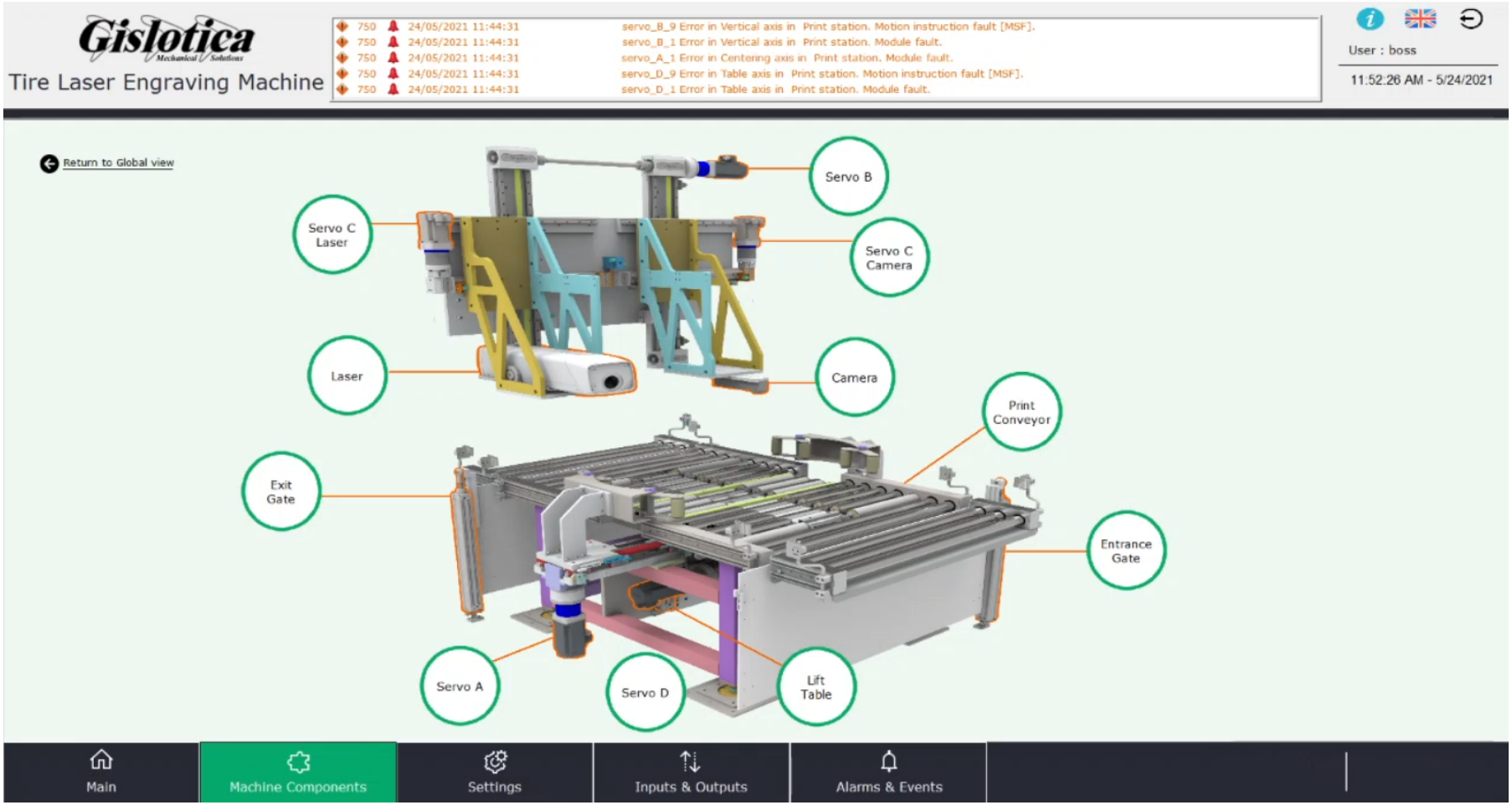

Gislotica
Tire Laser Engraving Machine
User : DESKTOP-C9N42J5\
11:47:08 AM - 5/24/2021
servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
servo_B_1 Error in Vertical axis in Print station. Module fault.
servo_A_1 Error in Centering axis in Print station. Module fault.
servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
servo_D_1 Error in Table axis in Print station. Module fault.
Infeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Print Station Status
In process
Pausing
Paused
Error code: 66125824
Process Flow
Abort tyre
Outfeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Initialization Routine
Initialized
Can initialize
In process
Start
Last 25 tyres
Tyre ID
Status
Filename
Time enter
Time
Tyres in Stations
Station
Tyre ID
Status
Mark Angle (deg)
Mark radius (mm)
Ext. Diam (mm)
Int. Diam (mm)
Width (mm)
Filename
Ref. Type
Ref. Radiu
Infeed
Print
Outfeed
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

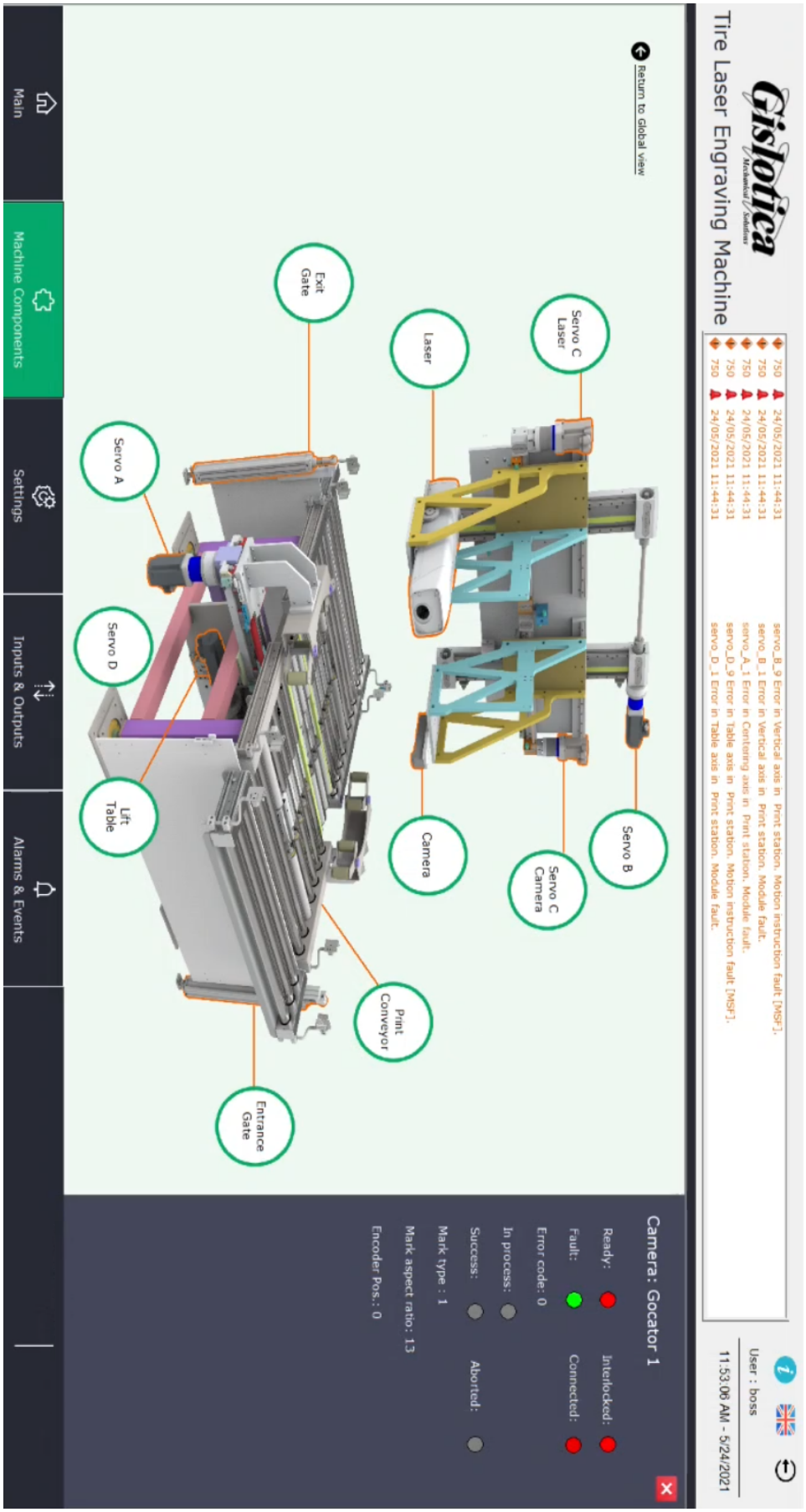

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:48:54 AM - 5/24/2021
Infeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Print Station Status
In process
Pausing
Paused
Error code: 66125824
Process Flow
Abort tyre
Outfeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Last 25 tyres
Tyre ID Status Filenam
4 Done mrk36
3 Done mrk12
2 Done mrk11
19 Unknown mrk20
18 Done mrk20
17 Done mrk20
16 Done mrk19
15 Done mrk11
14 Done mrk10
13 Done mrk10
12 Aborted mrk74
10 Aborted mrk57
8 Aborted mrk39
6 Aborted mrk28
4 Aborted mrk26
25 Aborted mrk10
5 Unknown mrk16
4 Done mrk19
Process Flow: Infeed Station
Receiving
Measuring tyre
Giving
Initialization Routine
Initialized
Can initialize
In process
Start
Tyres in Stations
Station Tyre ID Status Mark Angle (deg) Mark radius (mm) Ext. Diam (mm) Int. Diam (mm) Width (mm
Infeed 6 In process 221.0 200.0 235.0 190.0 190.0
Print 5 In process 221.0 200.0 235.0 190.0 190.0
Outfeed
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

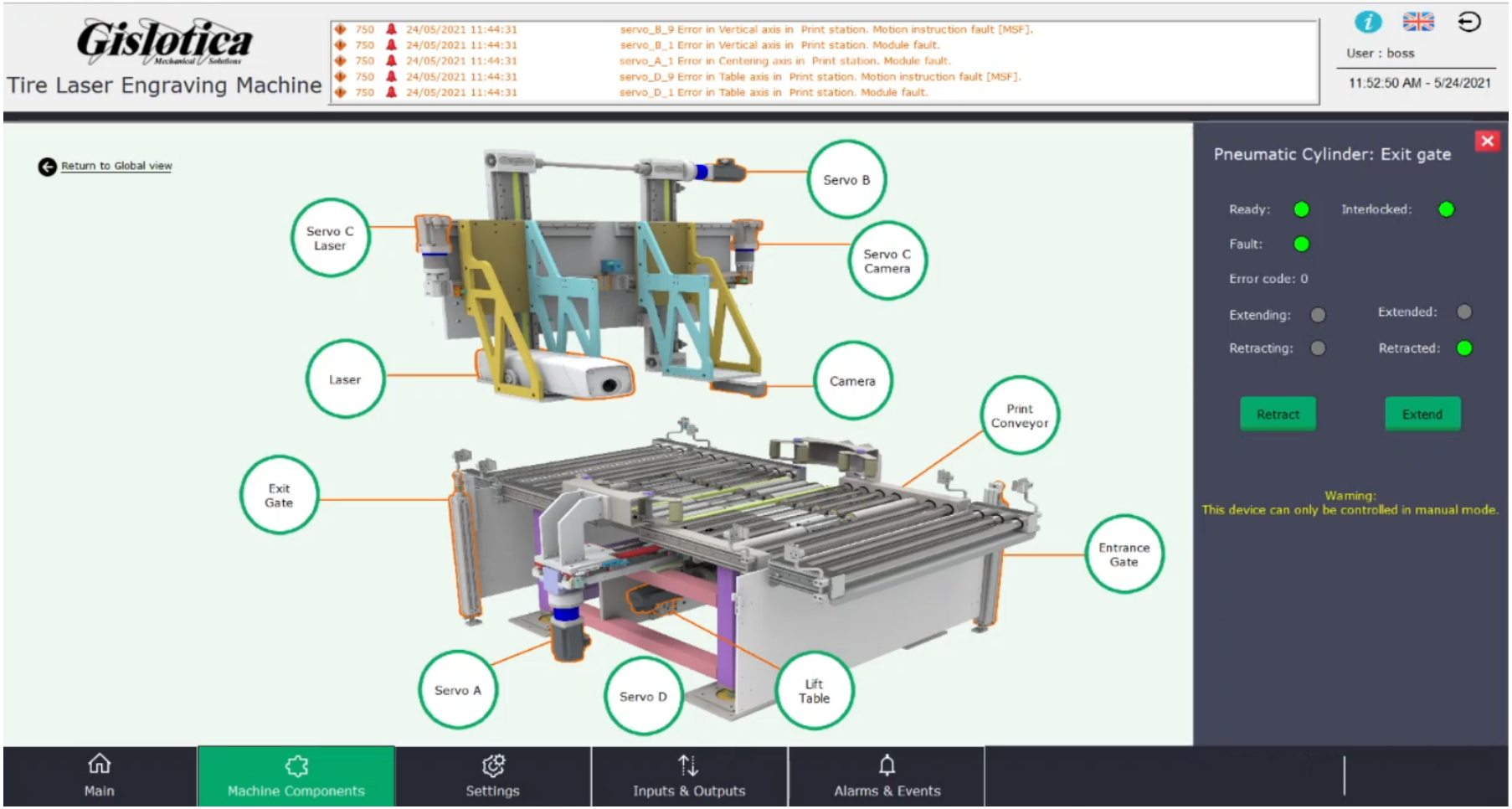

Gislotica
Tire Laser Engraving Machine
User : boss
11:49:05 AM - 5/24/2021
Infeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Print Station Status
In process
Pausing
Paused
Error code: 66125824
Process Flow
Abort tyre
Outfeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Initialization Routine
Initialized
Can initialize
In process
Start
Last 25 tyres
Tyre ID
Status
Tyres in Stations
Station
Tyre ID
Status
Mark Angle (deg)
Mark radius (mm)
Ext. Diam (mm)
Int. Diam (mm)
Width (mm
Infeed
6
In process
221.0
200.0
235.0
190.0
190.0
Print
5
In process
221.0
200.0
235.0
190.0
190.0
Outfeed
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Process Flow: Print Station
Receiving
Measuring tyre
Centering
Finding reference
Printing
Giving
Aborting

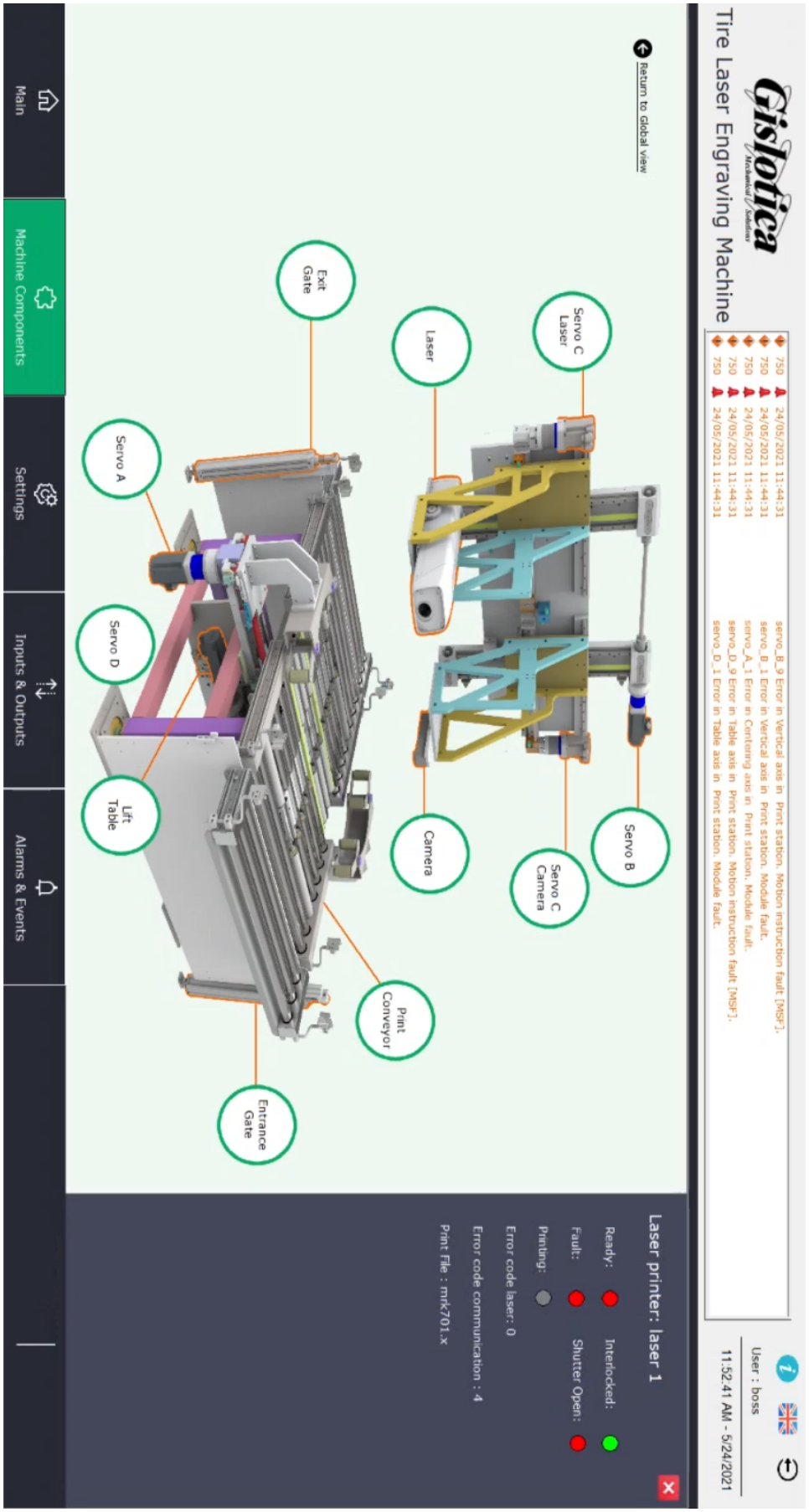

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:51:20 AM - 5/24/2021
Infeed Conveyor
Print Station
Outfeed Conveyor
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

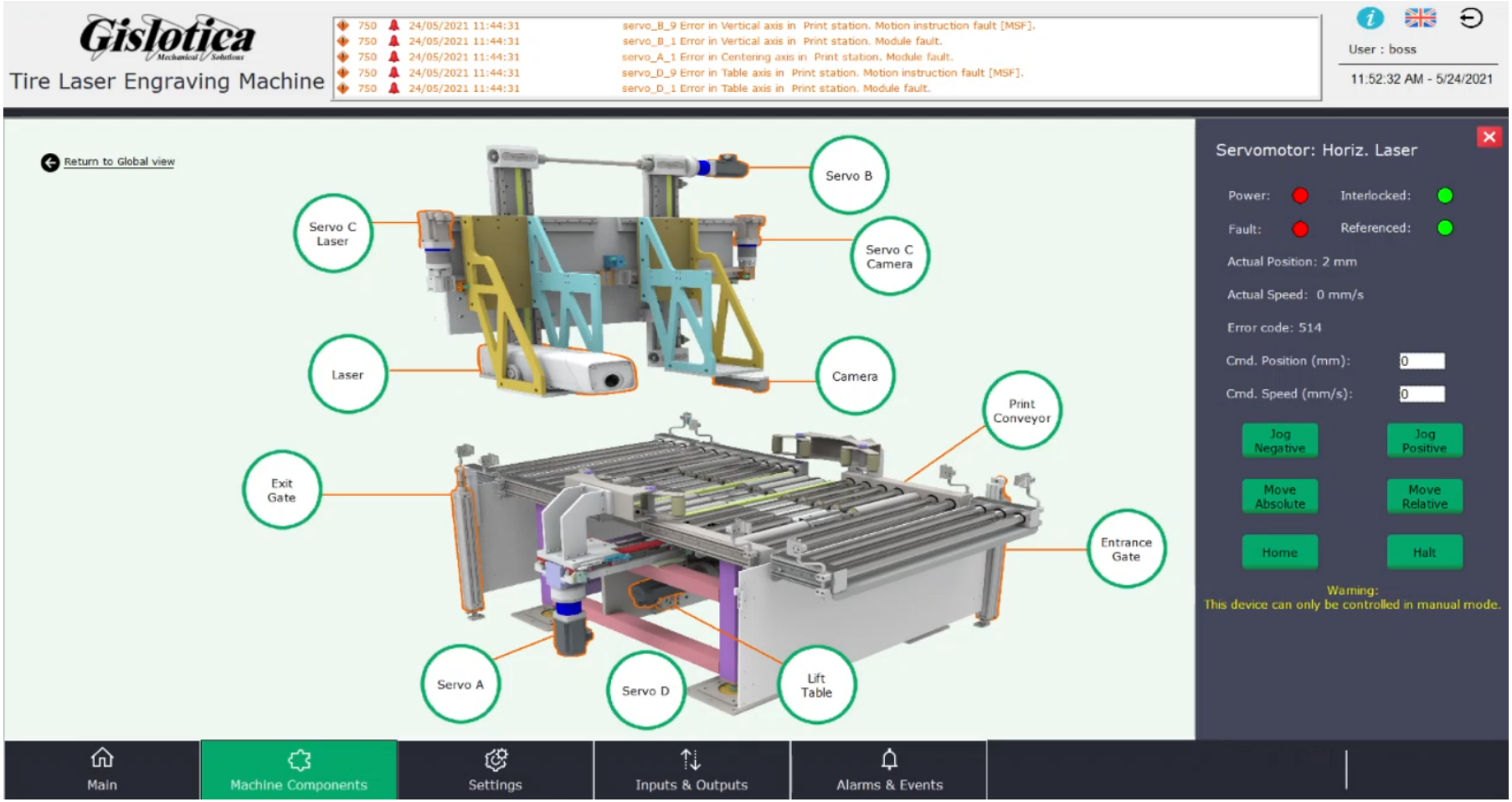

Gislotica
Tire Laser Engraving Machine
User : boss
11:51:47 AM - 5/24/2021
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Infeed Conveyor
Print Station
Conveyor: Infeed
Ready:
Interlocked:
Fault:
Commanded Speed: 0
Error code: 2304
Running Forward:
Running Backward:
Jog Negative
Jog Positive
Warning:
This device can only be controlled in manual mode.

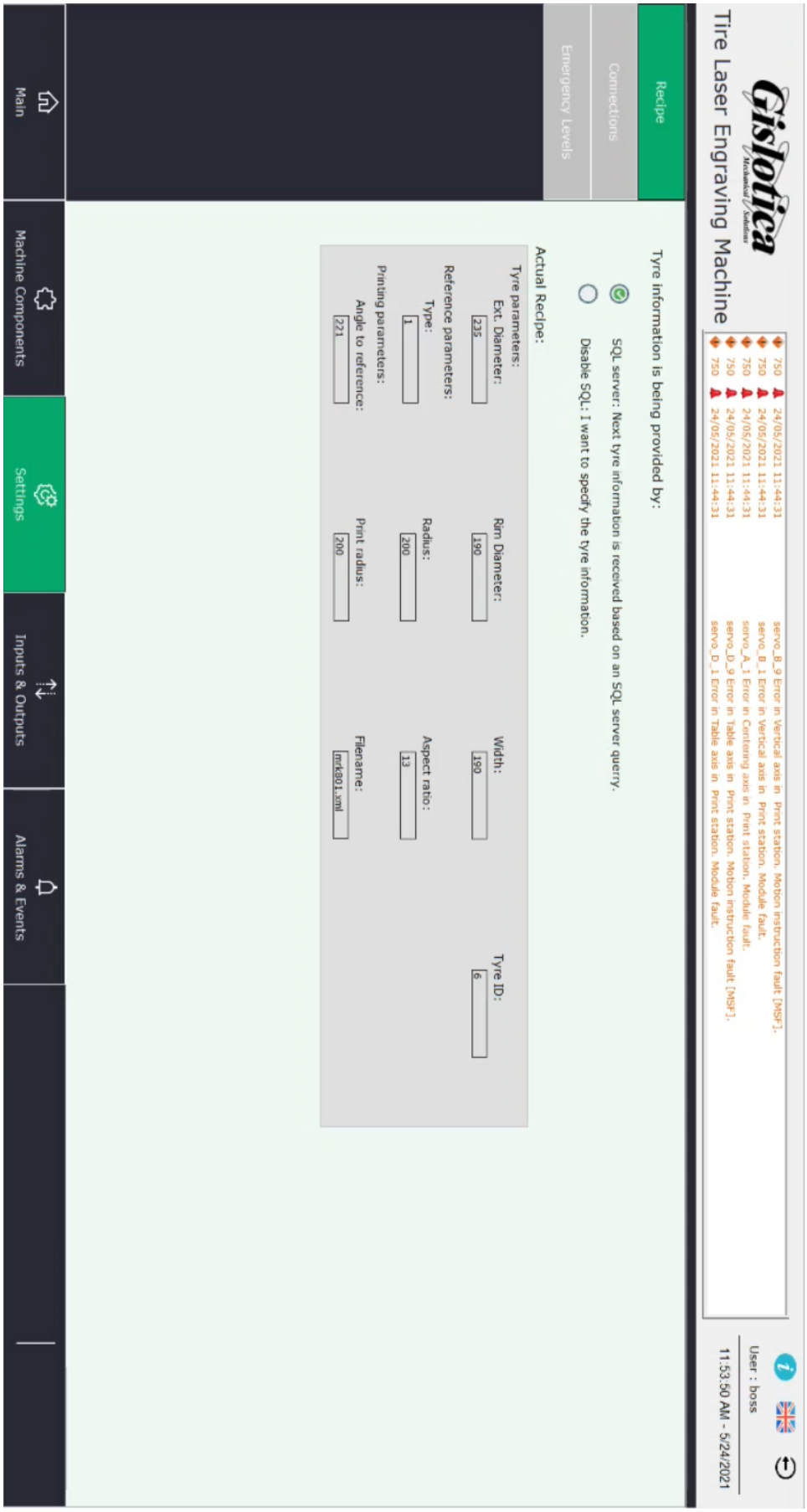

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:52.26 AM - 5/24/2021
Return to Global view
Servo B
Servo C Laser
Servo C Camera
Laser
Camera
Print Conveyor
Exit Gate
Entrance Gate
Servo A
Servo D
Lift Table
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

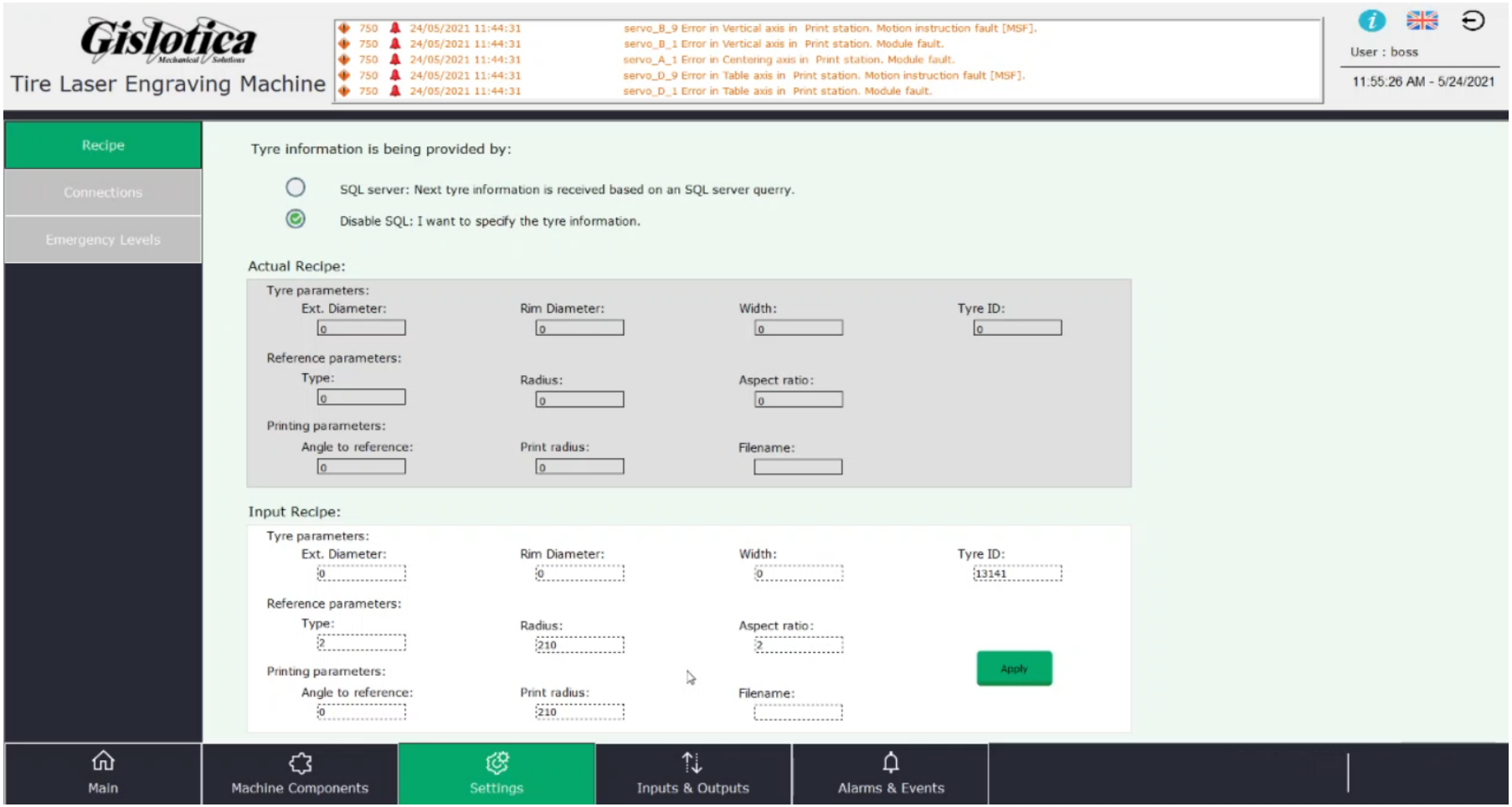

Gislotica
Tire Laser Engraving Machine
User : boss
11:53:06 AM - 5/24/2021
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Return to Global view
Servo C Laser
Laser
Exit Gate
Servo A
Servo D
Lift Table
Servo B
Servo C Camera
Camera
Print Conveyor
Entrance Gate
Camera: Gocator 1
Ready:
Interlocked:
Fault:
Connected:
Error code: 0
In process:
Success:
Aborted:
Mark type : 1
Mark aspect ratio: 13
Encoder Pos.: 0

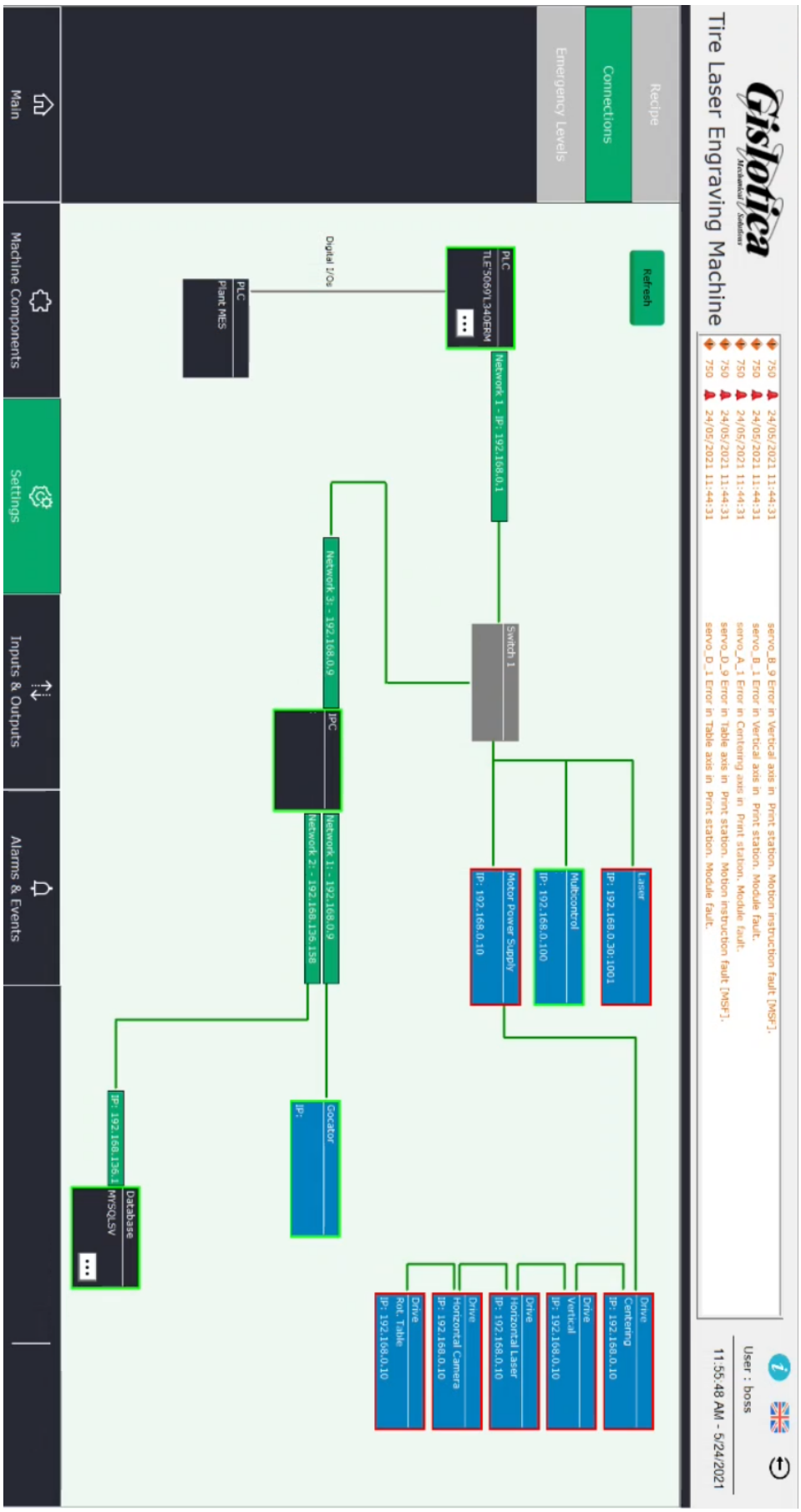

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:52:50 AM - 5/24/2021
Return to Global view
Servo B
Servo C Laser
Servo C Camera
Laser
Camera
Print Conveyor
Exit Gate
Entrance Gate
Servo A
Servo D
Lift Table
Pneumatic Cylinder: Exit gate
Ready:
Interlocked:
Fault:
Error code: 0
Extending:
Extended:
Retracting:
Retracted:
Retract
Extend
Warning:
This device can only be controlled in manual mode.
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

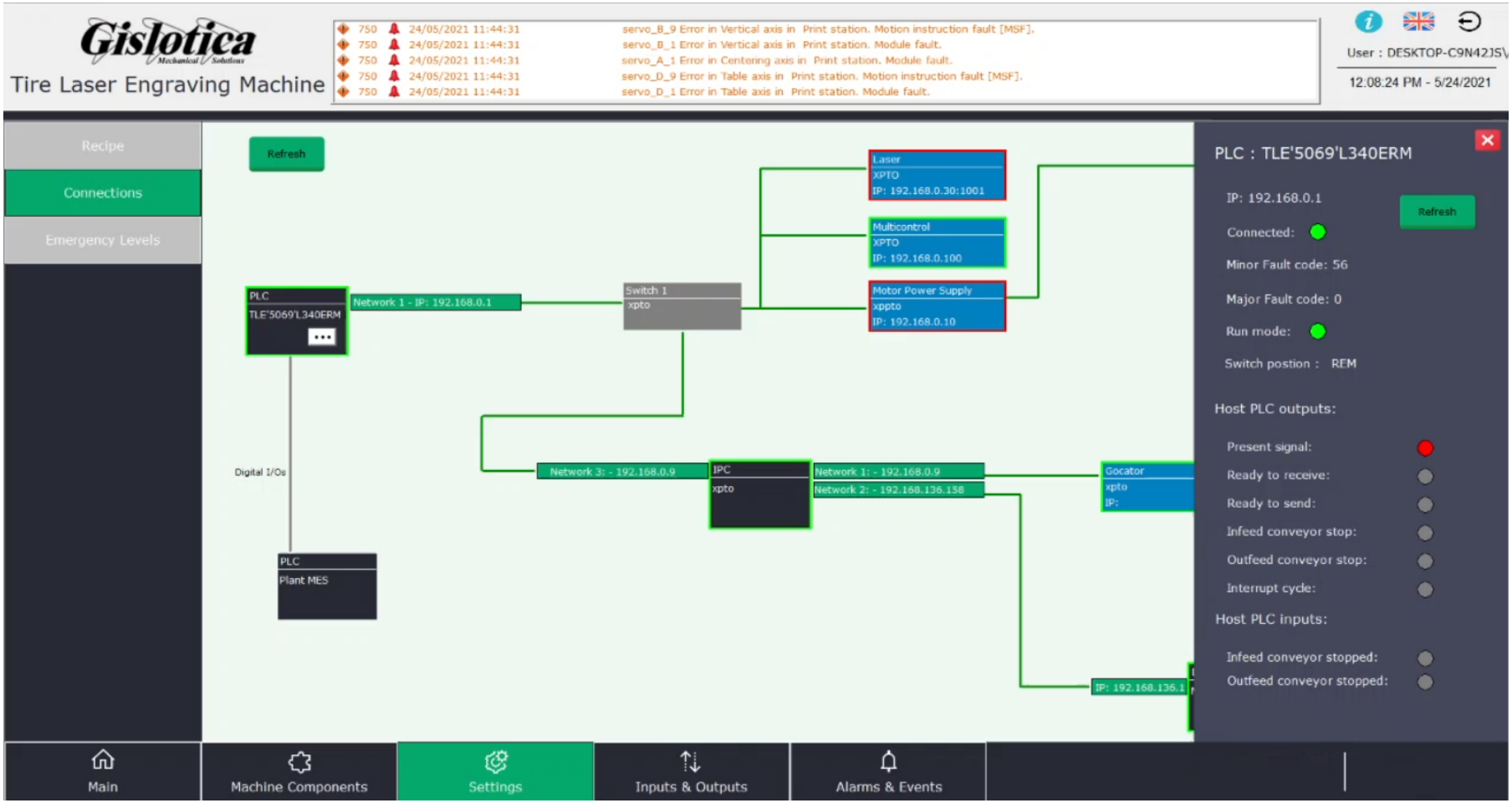

Gislotica
Tire Laser Engraving Machine
User : boss
11:52:41 AM - 5/24/2021
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Return to Global view
Servo C Laser
Laser
Exit Gate
Servo A
Servo D
Lift Table
Servo B
Servo C Camera
Camera
Print Conveyor
Entrance Gate
Laser printer: laser 1
Ready:
Interlocked:
Fault:
Shutter Open:
Printing:
Error code laser: 0
Error code communication : 4
Print File : mrk701.x

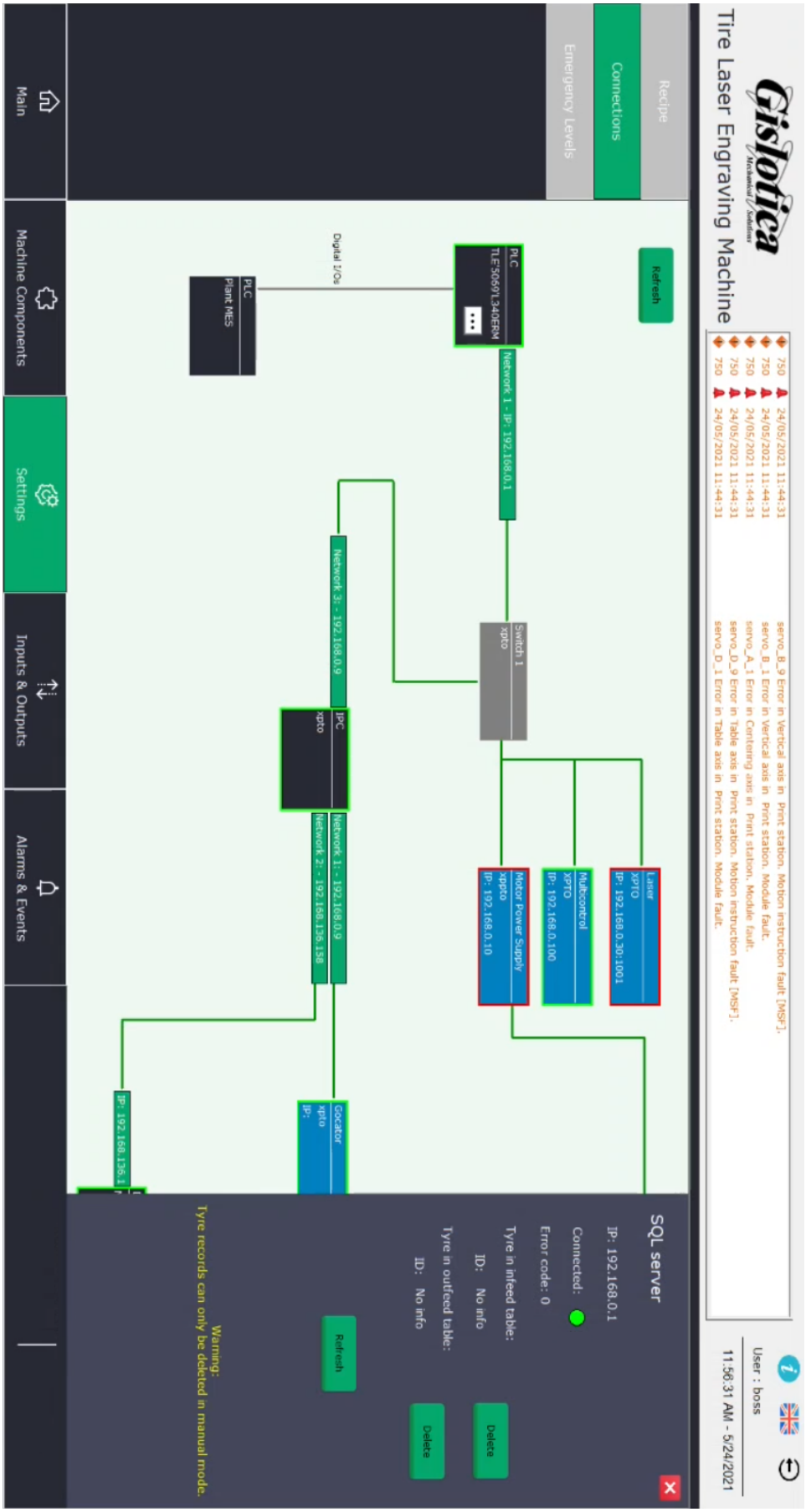

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:52:32 AM - 5/24/2021
Return to Global view
Servo B
Servo C Laser
Servo C Camera
Laser
Camera
Print Conveyor
Exit Gate
Entrance Gate
Servo A
Servo D
Lift Table
Servomotor: Horiz. Laser
Power:
Interlocked:
Fault:
Referenced:
Actual Position: 2 mm
Actual Speed: 0 mm/s
Error code: 514
Cmd. Position (mm): 0
Cmd. Speed (mm/s): 0
Jog Negative
Jog Positive
Move Absolute
Move Relative
Home
Halt
Warning:
This device can only be controlled in manual mode.
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

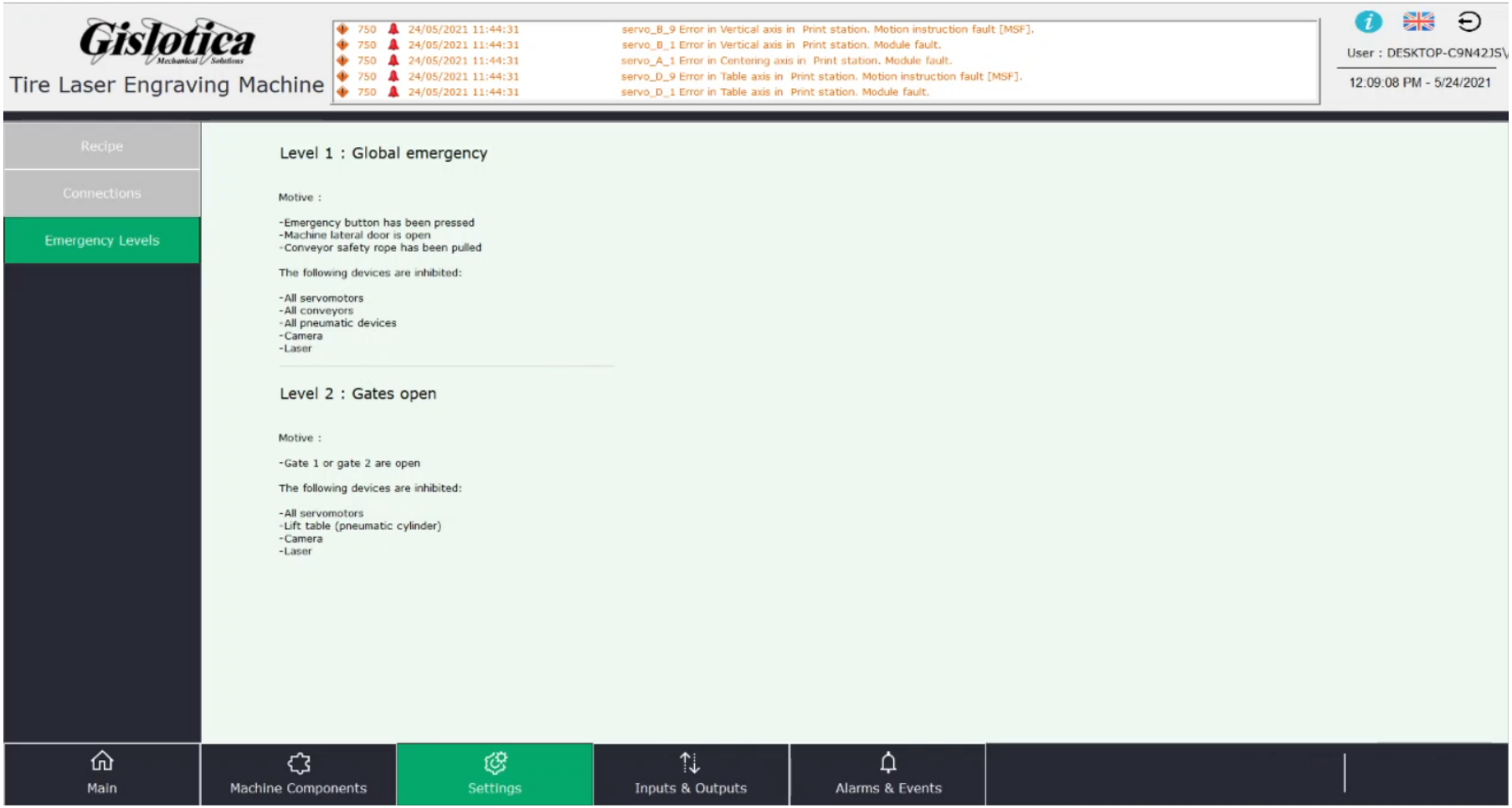

Gislotica
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:53:50 AM - 5/24/2021
Recipe
Connections
Emergency Levels
Tyre information is being provided by:
SQL server: Next tyre information is received based on an SQL server query.
Disable SQL: I want to specify the tyre information.
Actual Recipe:
Tyre parameters:
Ext. Diameter: 235
Rim Diameter: 190
Width: 190
Tyre ID: 6
Reference parameters:
Type: 1
Radius: 200
Aspect ratio: 13
Printing parameters:
Angle to reference: 221
Print radius: 200
Filename: mrk801.xml
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

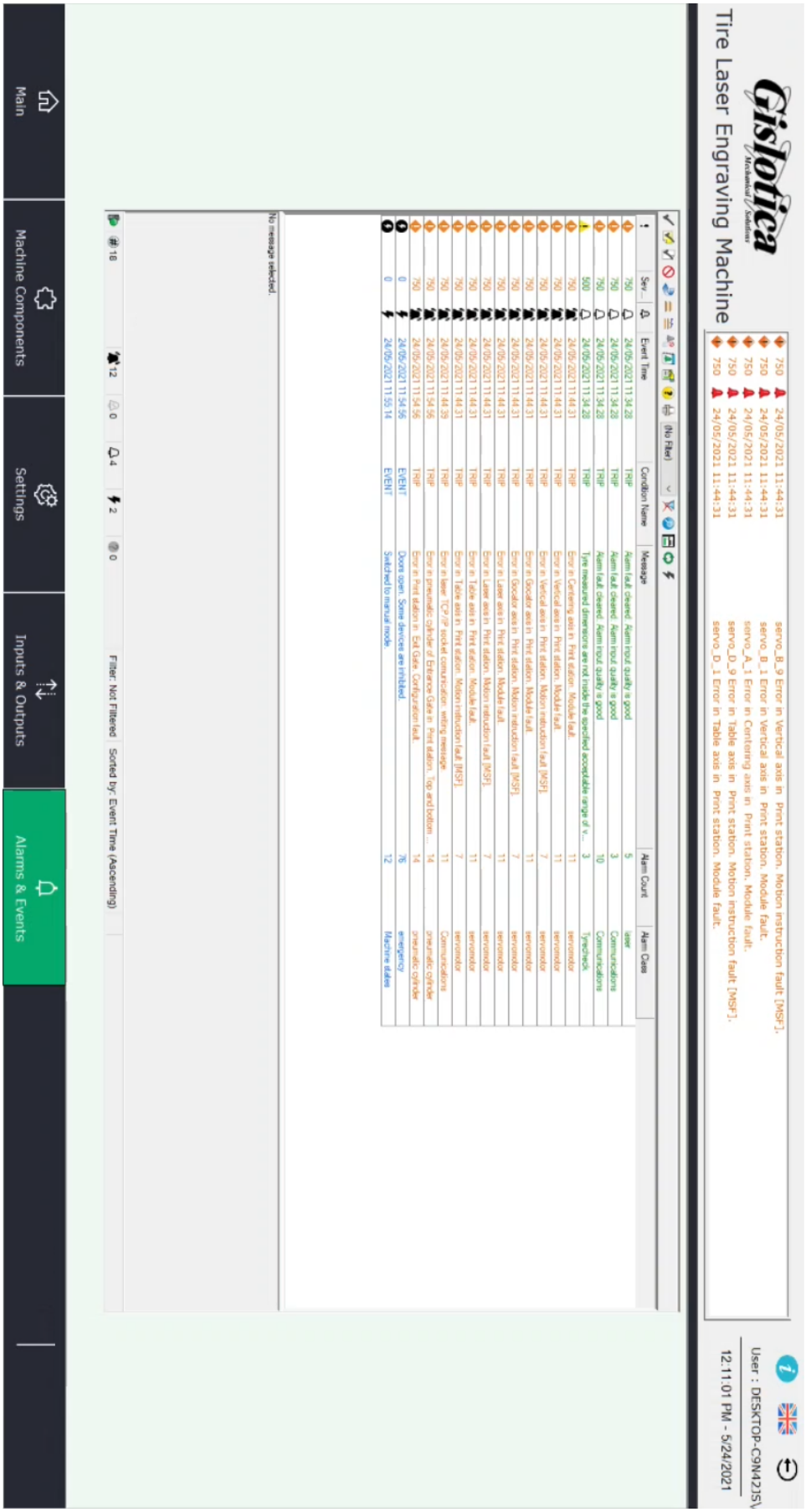

Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : boss
11:55:26 AM - 5/24/2021
Recipe
Connections
Emergency Levels
Tyre information is being provided by:
SQL server: Next tyre information is received based on an SQL server querry.
Disable SQL: I want to specify the tyre information.
Actual Recipe:
Tyre parameters:
Ext. Diameter:
Rim Diameter:
Width:
Tyre ID:
Reference parameters:
Type:
Radius:
Aspect ratio:
Printing parameters:
Angle to reference:
Print radius:
Filename:
Input Recipe:
Tyre parameters:
Ext. Diameter:
Rim Diameter:
Width:
Tyre ID:
13141
Reference parameters:
Type:
Radius:
210
Aspect ratio:
Apply
Printing parameters:
Angle to reference:
Print radius:
210
Filename:
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events

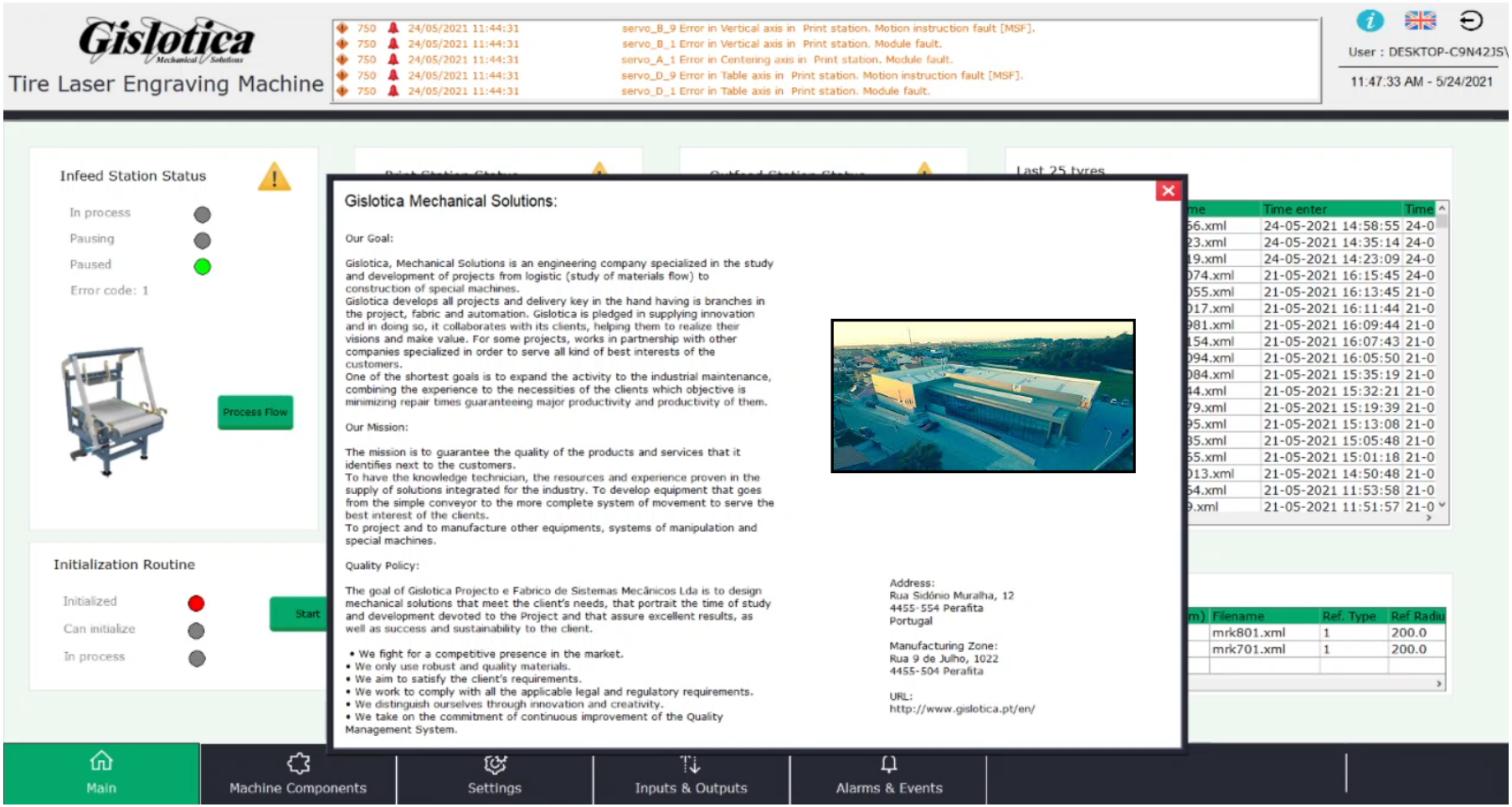

Gislotica
Tire Laser Engraving Machine
User : boss
11:55:48 AM - 5/24/2021
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Recipe
Connections
Emergency Levels
Refresh
PLC
TLE'S0601340FBM
PLC
Plant MES
Digital I/Os
Network 1 - IP: 192.168.0.1
Switch 1
Network 3: - 192.168.0.9
IPC
Network 1: - 192.168.0.9
Network 2: - 192.168.136.158
Laser
IP: 192.168.0.30:1001
Multicontrol
IP: 192.168.0.100
Motor Power Supply
IP: 192.168.0.10
Drive
Centering
IP: 192.168.0.10
Drive
Vertical
IP: 192.168.0.10
Drive
Horizontal Laser
IP: 192.168.0.10
Drive
Horizontal Camera
IP: 192.168.0.10
Drive
Rot. Table
IP: 192.168.0.10
Locator
IP:
IP: 192.168.136.1
Database
MYSQLSV


Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : DESKTOP-C9N42JS\
12:08:24 PM - 5/24/2021
Recipe
Connections
Emergency Levels
Refresh
Laser
XPTO
IP: 192.168.0.30:1001
Multicontrol
XPTO
IP: 192.168.0.100
Motor Power Supply
xppto
IP: 192.168.0.10
PLC
TLE'5069'L340ERM
Network 1 - IP: 192.168.0.1
Switch 1
xpto
Digital I/Os
Network 3: - 192.168.0.9
IPC
xpto
Network 1: - 192.168.0.9
Network 2: - 192.168.136.158
Gocator
xpto
IP:
IP: 192.168.136.1
PLC
Plant MES
PLC : TLE'5069'L340ERM
IP: 192.168.0.1
Refresh
Connected:
Minor Fault code: 56
Major Fault code: 0
Run mode:
Switch postion : REM
Host PLC outputs:
Present signal:
Ready to receive:
Ready to send:
Infeed conveyor stop:
Outfeed conveyor stop:
Interrupt cycle:
Host PLC inputs:
Infeed conveyor stopped:
Outfeed conveyor stopped:
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events


Gislotica
Tire Laser Engraving Machine
User : boss
11:56:31 AM - 5/24/2021
Recipe
Connections
Emergency Levels
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Refresh
PLC
TLE 30691.340ERM
Network 1 - IP: 192.168.0.1
Digital I/Os
PLC
Plant MES
Switch 1
xpto
Laser
XPTO
IP: 192.168.0.30:1001
Multicontrol
XPTO
IP: 192.168.0.100
Motor Power Supply
xpto
IP: 192.168.0.10
Network 3: - 192.168.0.9
IPC
xpto
Network 1: - 192.168.0.9
Network 2: - 192.168.136.158
Locator
xpto
IP:
IP: 192.168.136.1
SQL server
IP: 192.168.0.1
Connected:
Error code: 0
Tyre in infeed table:
ID: No info
Delete
Tyre in outfeed table:
ID: No info
Delete
Refresh
Warning:
Tyre records can only be deleted in manual mode.


Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : DESKTOP-C9N42JS\
12:09:08 PM - 5/24/2021
Recipe
Connections
Emergency Levels
Level 1 : Global emergency
Motive :
-Emergency button has been pressed
-Machine lateral door is open
-Conveyor safety rope has been pulled
The following devices are inhibited:
-All servomotors
-All conveyors
-All pneumatic devices
-Camera
-Laser
Level 2 : Gates open
Motive :
-Gate 1 or gate 2 are open
The following devices are inhibited:
-All servomotors
-Lift table (pneumatic cylinder)
-Camera
-Laser
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events


Gislotica
Tire Laser Engraving Machine
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events
Event Time
Condition Name
Message
Alarm Count
Alarm Class
No message selected.
Filter: Not Filtered
Sorted by: Event Time (Ascending)
User : DESKTOP-C9N42JS
12:11:01 PM - 5/24/2021


Gislotica
Mechanical Solutions
Tire Laser Engraving Machine
750 24/05/2021 11:44:31 servo_B_9 Error in Vertical axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_B_1 Error in Vertical axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_A_1 Error in Centering axis in Print station. Module fault.
750 24/05/2021 11:44:31 servo_D_9 Error in Table axis in Print station. Motion instruction fault [MSF].
750 24/05/2021 11:44:31 servo_D_1 Error in Table axis in Print station. Module fault.
User : DESKTOP-C9N42JS\
11:47:33 AM - 5/24/2021
Infeed Station Status
In process
Pausing
Paused
Error code: 1
Process Flow
Initialization Routine
Initialized
Can initialize
In process
Start
Last 25 tyres
Time enter
24-05-2021 14:58:55
24-05-2021 14:35:14
24-05-2021 14:23:09
21-05-2021 16:15:45
21-05-2021 16:13:45
21-05-2021 16:11:44
21-05-2021 16:09:44
21-05-2021 16:07:43
21-05-2021 16:05:50
21-05-2021 15:35:19
21-05-2021 15:32:21
21-05-2021 15:19:39
21-05-2021 15:13:08
21-05-2021 15:05:48
21-05-2021 15:01:18
21-05-2021 14:50:48
21-05-2021 11:53:58
21-05-2021 11:51:57
Filename
Ref. Type
Ref Radiu
mrk801.xml 1 200.0
mrk701.xml 1 200.0
Gislotica Mechanical Solutions:
Our Goal:
Gislotica, Mechanical Solutions is an engineering company specialized in the study and development of projects from logistic (study of materials flow) to construction of special machines.
Gislotica develops all projects and delivery key in the hand having is branches in the project, fabric and automation. Gislotica is pledged in supplying innovation and in doing so, it collaborates with its clients, helping them to realize their visions and make value. For some projects, works in partnership with other companies specialized in order to serve all kind of best interests of the customers.
One of the shortest goals is to expand the activity to the industrial maintenance, combining the experience to the necessities of the clients which objective is minimizing repair times guaranteeing major productivity and productivity of them.
Our Mission:
The mission is to guarantee the quality of the products and services that it identifies next to the customers.
To have the knowledge technician, the resources and experience proven in the supply of solutions integrated for the industry. To develop equipment that goes from the simple conveyor to the more complete system of movement to serve the best interest of the clients.
To project and to manufacture other equipments, systems of manipulation and special machines.
Quality Policy:
The goal of Gislotica Projecto e Fabrico de Sistemas Mecânicos Lda is to design mechanical solutions that meet the client's needs, that portrait the time of study and development devoted to the Project and that assure excellent results, as well as success and sustainability to the client.
• We fight for a competitive presence in the market.
• We only use robust and quality materials.
• We aim to satisfy the client's requirements.
• We work to comply with all the applicable legal and regulatory requirements.
• We distinguish ourselves through innovation and creativity.
• We take on the commitment of continuous improvement of the Quality Management System.
Address:
Rua Sidónio Muralha, 12
4455-554 Perafita
Portugal
Manufacturing Zone:
Rua 9 de Julho, 1022
4455-504 Perafita
URL:
http://www.gislotica.pt/en/
Main
Machine Components
Settings
Inputs & Outputs
Alarms & Events