



**André
Gordo
Saraiva**

**Estágio de Projeto Elétrico, Automação e Robótica
Industrial
(Linha de Montagem de Bombas de Óleo de Caudal
Variável)**



**André
Gordo
Saraiva**

**Estágio de Projeto Elétrico, Automação e Robótica
Industrial
(Linha de Montagem de Bombas de Óleo de Caudal
Variável)**

Relatório de estágio apresentado à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia em Automação Industrial, realizada sob a orientação científica do Doutor Pedro Nicolau Faria da Fonseca, Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro e sob co-orientação do Engenheiro Abílio Manuel Ribeiro Borges, Assistente Convidado do Departamento de Engenharia Mecânica da Universidade de Aveiro

O júri / the jury

Presidente / president

Prof. Doutor José Paulo Oliveira Santos
Professor Auxiliar do Departamento de Engenharia Mecânica da Universidade de Aveiro

Vogais / examiners committee

Prof. Doutor Paulo Augusto Ferreira de Abreu
Professor Auxiliar do Departamento de Engenharia Mecânica da Faculdade de Engenharia da Universidade do Porto

Prof. Doutor Pedro Nicolau Faria da Fonseca
Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Agradecimentos / acknowledgements

Gostaria, antes de mais, de agradecer à Atena – Automação Industrial, Lda. pela grande oportunidade de realização de um estágio no seu departamento de automação, pois tornou este trabalho possível. Quero também agradecer à fantástica equipa que forma a empresa, pelo imenso apoio técnico e humano e ajuda de valor incalculável, nomeadamente ao Carlos, ao Diogo, ao Luís, ao Manuel e ao Rúben. Os meus sinceros agradecimentos. Finalmente, um enorme obrigado aos mais próximos, por tudo. És maravilhosa, Ana.

Palavras-chave

automação, robótica, visão por computador, projecto eléctrico, indústria, programação

Resumo

O presente trabalho trata do desenvolvimento de sistemas automáticos para a indústria, em concreto de sistemas integrantes de uma linha de montagem de bombas de óleo de caudal variável. O principal objectivo foi a aplicação de conhecimentos desenvolvidos durante o percurso académico e desenvolvimento de novas aprendizagens úteis para a vida profissional futura, dentro da área de automação, através da realização dos sistemas acima mencionados com sucesso. Com o suporte de uma equipa profissional, foram desenvolvidos vários sistemas, nomeadamente sistemas de visão para inspecção da boa montagem de peças por equipamentos automáticos, um posto de trabalho com componente automática e de interacção com um operador, integrando um braço robótico colaborativo e partes de um sistema de gestão de produção. A grande diversidade de trabalho desenvolvido revelou-se uma grande fonte de conhecimentos e técnicas de desenvolvimento de sistemas automáticos. Os sistemas desenvolvidos tiveram também um desempenho positivo no seu funcionamento.

Keywords

Automation, robotics, computer vision, electric project, industry, programming

Abstract

This work presents the development of automated systems for the industry, more precisely systems that are part of a variable flow oil pumps' assembly line. The primary objective was the application of knowledge acquired during the academic path and the development of new learnings useful for a future professional life in the area of automation, by successfully creating the systems previously mentioned. With the support of a professional team, several systems were developed, including computer vision systems for the inspection of automatically assembled parts, a workstation with both automatic and manual operated parts featuring a collaborative robotic arm, and sections of a production management system. The diversity of work done revealed itself as a good source of knowledge and techniques regarding the development of automatic systems. The developed systems also had a good performance, regarding their operation.

ÍNDICE

ÍNDICE	i
ÍNDICE DE FIGURAS	v
ÍNDICE DE TABELAS	ix
ACRÓNIMOS E ABREVIATURAS	xi
1. INTRODUÇÃO	1
1.1. Apresentação do Estágio e Motivações	1
1.2. Bombas de Óleo de Caudal Variável	2
1.3. Linha de Montagem de Bombas de Óleo de Caudal Variável	5
1.4. Objectivos do Estágio	13
2. ESTADO DA ARTE	15
2.1. Introdução	15
2.2. Projecto Eléctrico	15
2.3. Projecto de Automação	17
2.4. Padrões de Programação e Análise de Programas	27
3. FERRAMENTAS E DISPOSITIVOS	29
3.1. Introdução	29
3.2. Dispositivos e Equipamentos	29
3.3. <i>Software</i> Utilizado	31
3.4. Interfaces de Comunicação	36
4. SISTEMAS DE VISÃO	39
4.1. Introdução	39
4.2. Requisitos do Sistema	41
4.3. Esquema Eléctrico do Sistema de Visão	42

4.4. Ajuste Mecânico do Sistema	43
4.5. Programa de Inspeção de Bombas de Óleo	47
4.6. Programa do Autómato	57
4.7. Segundo Sistema de Visão	59
4.8. Resultados	61
4.9. Conclusões	63
5. SISTEMA DE GESTÃO DE PRODUÇÃO	65
5.1. Introdução	65
5.2. Funções Desenvolvidas	66
5.3. Resultados	78
5.4. Conclusões	78
6. POSTO DE TRABALHO OP25	81
6.1. Introdução	81
6.2. Programa do Autómato	83
6.3. Programa do Manipulador Robótico	91
6.4. Parametrizações de Outros Dispositivos	91
6.5. Interfaces Homem-Máquina	95
6.6. Resultados	99
6.7. Conclusões	101
7. OUTROS TRABALHOS DESENVOLVIDOS	103
7.1. Esquemas Eléctricos dos Robôs dos OP90.2 e OP90.3	103
7.2. Esquema Eléctrico do Bastidor do Servidor	105
7.3. HMI's dos OP90 e OP40	106
8. CONCLUSÕES	109
8.1. Considerações Finais	109
8.2. Trabalho Futuro	110
9. BIBLIOGRAFIA	111
ANEXOS	

ÍNDICE DE FIGURAS

1.1.	Bomba de óleo de caudal variável (R9M, montagem parcial), vista geral	3
1.2.	Bomba de óleo de caudal variável (R9M, montagem parcial), vista de cima	4
1.3.	Palhetas polidas e escuras	4
1.4.	Linha de montagem de bombas de óleo de caudal variável: esquema geral, com postos de trabalho identificados	7
1.5.	Paquete de bombas Hxx entre o controlo de pinhão e o OP90	9
1.6.	Diagrama de blocos da linha de montagem de bombas de óleo	12
1.7.	Diagrama de blocos de um posto de trabalho genérico	13
2.1.	Hierarquia de controlo de um sistema fabril	18
2.2.	Esquemática de um PLC e dispositivos periféricos	19
2.3.	Comparação entre LAD, FDB e IL, utilizando um “AND” e um “OR” lógicos e um circuito de retorno	20
2.4.	Exemplo de programa em ST	21
2.5.	Exemplo de programa utilizando SFC	21
2.6.	Exemplo de interface homem-máquina	23
3.1.	Siemens SIMATIC S7-300 CPU 315-2 PN/DP	30
3.2.	Teledyne Dalsa BOA BVS-1280M-INS	30
3.3.	Robô ABB IRB 2600	31
3.4.	Robô Universal Robots UR10	31
3.5.	Ambiente gráfico do TIA Portal V12	32
3.6.	Ambiente gráfico do STEP 7 V5.5, janela de projecto	33
3.7.	Ambiente gráfico do WinCC V7.3	33

3.8.	Ambiente gráfico do EPlan Electric P8	34
3.9.	Ambiente de desenvolvimento do iNspect Express. Simulação a correr	35
3.10.	Ambiente gráfico do RobotStudio. Célula de fabrico vazia com robô IRB 1600 modelado	35
3.11.	Ambiente gráfico do DSM MultiControl4	36
4.1.	Esquema do posto de trabalho OP70.5, vista isométrica	40
4.2.	Esquema do OP70.5, vista lateral, com bomba posicionada (dentro do quadrado)	40
4.3.	Anilha, palhetas e segmento	41
4.4.	Esquema das conexões da câmara Dalsa BOA	42
4.5.	Esquema do posicionamento do sistema de visão no posto OP70.5, visto de cima	43
4.6.	Posicionamento da iluminação no sistema de visão	45
4.7.	Estudo do tempo de exposição e brilho	46
4.8.	Exemplo de imagem captada após ajuste óptico e de posição do sistema	46
4.9.	Gradientes de interesse para localizador	48
4.10.	Área de interesse para localizador	48
4.11.	Análise da existência de anilha na bomba de óleo	50
4.12.	ROI de identificação de palhetas, com sete identificadas	51
4.13.	Topos de uma selecção de palhetas	52
4.14.	Máscara de procura do segmento, com limites identificados	53
4.15.	Análise realizada a uma bomba de óleo	53
4.16.	Inicialização da solução	55
4.17.	Script executado a cada 10 ms	55
4.18.	Script executado após realizada cada análise	56
4.19.	Diagrama de actividade do programa de PLC do sistema de visão	58
4.20.	Exemplo de imagem captada pelo segundo sistema de visão	60

4.21.	Exemplo de análise efectuada no segundo sistema de visão	60
5.1.	Página principal do sistema de gestão e produção da linha de montagem	66
5.2.	Diagrama de checkLogin()	68
5.3.	Diagrama de logInOut()	68
5.4.	Diagrama de exitRuntime()	68
5.5.	Diagrama de OPtoNOP()	69
5.6.	Diagrama de NOPtoOP()	69
5.7.	Caixa de diálogo de abertura de ficheiro	70
5.8.	Diagrama de fileDialog()	70
5.9.	Janela de importação e exportação do sistema	71
5.10.	Diagrama de uaExportEntries()	72
5.11.	Diagrama de uaImportEntries()	72
5.12.	Diagrama de pumpSearchImport()	74
5.13.	Diagrama de fileSearch()	74
5.14.	Diagrama de uaEditID()	74
5.15.	Janela de produção semanal	75
5.16.	Diagrama de updateProdGraf()	76
5.17.	Diagrama de uaGetParts()	76
5.18.	Diagrama de writeLog()	77
6.1.	Componentes a montar pelo posto OP25, para modelo Hxx	82
6.2.	Esquema do posto de trabalho OP25, com transportador	82
6.3.	Relação entre UDT, DB e FC	84
6.4.	Diagrama de implantação do hardware do posto de trabalho OP25	86
6.5.	Diagrama de actividade da activação de um cilindro pneumático	88
6.6.	Exemplo de realimentação em Ladder	89
6.7.	Controlador de motores à direita, com fonte de alimentação do motor à esquerda	92

6.8.	Diagrama de actividade de rotina da prensa eléctrica	94
6.9.	Gráfico X-Y dos resultados da prensagem e um casquilho	95
6.10.	Esquema de menus e janelas do HMI do OP25	97
6.11.	Janela de modo automático da HMI do OP25	98
6.12.	Segunda janela de controlo manual da secção de prensagem de casquilhos	99
6.13.	Resultado das operações do posto de trabalho OP25	100
7.1.	Esquema eléctrico das conexões de sensores e válvulas do robô OP90.2	104
7.2.	Alimentação dos dispositivos do bastidor do servidor	105
7.3.	Esquema do bastidor, computador e <i>switch</i>	106
7.4.	Janela de modo automático do posto OP40	107

ÍNDICE DE TABELAS

4.1.	Combinações de bombas e paletes usadas em testes	61
4.2.	Resultados dos testes efectuados no sistema de visão do OP70.5	62
4.3.	Resultados das análises no segundo sistema de visão	63

ACRÓNIMOS E ABREVIATURAS

ACK	<i>Acknowledge</i> (Reconhecimento)
CENELEC	<i>Comité Européen de Normalisation Électrotechnique</i>
DB	<i>Data Block</i> (Bloco de dados)
CPU	<i>Central Processing Unit</i> (Unidade central de processamento)
ERP	<i>Enterprise Resource Planning</i> (Planeamento de recursos da empresa)
FB	<i>Function Block</i> (Bloco de funções)
FBD	<i>Functional Block Diagrams</i>
FC	<i>Function</i> (Função)
GSD	<i>Generic Station Description</i> (Descrição geral da estação)
HMI	<i>Human-Machine Interface</i> (Interface homem-máquina)
IEC	<i>International Electrical Commission</i>
IL	<i>Instruction List</i>
LAD	<i>Ladder</i>
MES	<i>Manufacturing Execution System</i> (Sistema de execução de manufactura)

OP	Operação
PLC	<i>Programmable Logic Computer</i> (Computador de lógica programável)
RFID	<i>Radio-Frequency Identification</i> (Identificação por radio-frequências)
ROI	<i>Region of Interest</i> (Região de interesse)
RTU	<i>Remote Terminal Unit</i> (Unidade terminal remota)
SCADA	<i>Supervisory Control and Data Acquisition</i>
SFC	<i>Sequential Function Chart</i> (Supervisão, controlo e aquisição de dados)
SQL	<i>Structured Query Language</i>
ST	<i>Structured Text</i>
UDT	<i>User Defined Datatype</i> (Tipo de dados definidos pelo utilizador)
UML	<i>Unified Modeling Language</i>

1. INTRODUÇÃO

1.1. Apresentação do Estágio e Motivações

A aliança entre sistemas de produção e a sua automatização permitiu à indústria a capacidade de produzir mais, melhor, em menos tempo e com custos menores. Levou também, naturalmente, à necessidade de mão de obra qualificada para conceptualizar, desenvolver e instalar tais sistemas automatizados nos locais de produção. Esta mão de obra necessita de possuir conhecimentos abrangentes, desde conceitos de mecânica, a capacidade de análise e projecto de sistemas eléctricos, assim como para programar sistemas electrónicos e automáticos e suas comunicações, até conceitos de *design* de produto e ergonomia, já que estes sistemas necessitam de ser manipulados, seja para efeitos de operação ou manutenção.

Um estágio curricular numa empresa de automação industrial permite o contacto directo com as mais recentes tecnologias utilizadas para a criação de sistemas de produção automatizados avançados para a indústria, assim como toda a problemática associada aos seus projectos. Comparando os objectivos do Mestrado em Engenharia de Automação Industrial – proporcionar formação técnica avançada no âmbito de sistemas automáticos, incluindo concepção e desenvolvimento de máquinas automatizadas e sua integração com outros sistemas automáticos [1] – com um estágio do género descrito acima, fica evidente a ligação entre este último e o curso: o estágio curricular oferece a possibilidade de complementar saberes anteriores e de realizar novas aprendizagens, tanto teóricas como práticas, dentro do meio. É também um veículo para a interacção com o mundo da indústria, sendo possível a visualização deste sector importante em termos de peso na economia a nível nacional [2] e mundial [3] a partir do seu interior. Resumidamente, é uma forma de obter experiência profissional antes mesmo de abandonar a vida académica.

O estágio ao qual este documento se refere envolve a integração numa equipa de desenvolvimento de um projecto de uma linha de montagem, na empresa “Atena – Automação Industrial, Lda.”, capaz de realizar a assemblagem de componentes, de modo a obter vários modelos de bombas de óleo de caudal variável para a indústria automóvel. Esta linha de montagem é destinada a trabalhar em paralelo com uma linha de montagem já instalada no cliente, de maneira a aumentar a produção total deste.

A Atena – Automação Industrial, Lda. desenvolve e realiza projectos de equipamentos específicos de acordo com as especificações dos clientes principalmente na área de automação industrial, serviços e áreas alimentares e biomédicas. A preocupação principal é a satisfação das necessidades desses clientes, valorizando a qualidade, inovação e fiabilidade dos projectos e equipamentos. A empresa construiu uma equipa altamente qualificada nas diversas áreas de actuação. A cultura de participação e complementaridade que procurou promover no seio dela, assente num diálogo permanente, procura estimular, motivar e desafiar para um desempenho individual e em equipa capaz de transformar as dificuldades em oportunidades. São disponibilizadas soluções em várias áreas, como sistemas automáticos e equipamentos especiais (sistemas de montagem e sistemas de transporte, por exemplo), robótica convencional e colaborativa, sistemas de visão artificial, meios de controlo dimensionais, informática industrial e supervisão, reconstrução de equipamentos e sistemas de gestão de energia, entre outros.

A não inclusão, neste documento, de explicações pormenorizadas de certas partes do projecto, equipamentos e programas desenvolvidos prende-se com a necessidade de se manter a confidencialidade do projecto, devido a acordos entre o cliente e a Atena – Automação Industrial, Lda.

1.2. Bombas de Óleo de Caudal Variável

As bombas de óleo que irão ser produzidas pela linha de montagem possuem, como principal característica, a capacidade de variar o caudal de óleo bombeado em função da velocidade de rotação do motor do automóvel. Esta variação permite a utilização de pressões de óleo inferiores no geral, comparativamente a bombas de óleo de caudal fixo, não afectando, porém, o desempenho a nível de potência do veículo. Isto leva a que o consumo energético da bomba seja menor, traduzindo-se num consumo de combustível reduzido comparativamente às bombas de óleo tradicionais. No entanto, a temperatura do óleo é também inferior, o que pode levar a emissões de NO_x – termo genérico para óxido e dióxido de azoto – ligeiramente superiores (aumentos de emissões esses marcadamente inferiores quando comparados com aumentos de emissões causados por outros sistemas de redução de consumo de combustível) [4].

Estas bombas de óleo são formadas por uma carcaça, por onde circula o óleo. Este óleo entra num sistema de compressão, constituído por um estator e rotor com palhetas que aumenta a pressão do líquido, sendo expelido para outra zona da carcaça, saindo depois da bomba de óleo. O estator é na verdade um elemento móvel, relativamente à carcaça, o que faz com que a geometria entre si e o rotor se altere, permitindo assim a variabilidade de geometria necessária à variação de caudal.

Existem três modelos de bombas de óleo a montar pela linha de montagem: Hxx, M9T e R9M. Diferenças entre estas incluem carcaças de geometrias diferentes, válvulas e sistemas de variação da posição do estator com componentes díspares e pinhões de dimensões diferentes. O modelo Hxx também inclui uma válvula electromecânica, inexistente nos outros modelos. Os rotores, estatores e palhetas são idênticos em todos os modelos, dimensionalmente. Adicionalmente, as palhetas do modelo M9T têm um acabamento escuro e fosco, comparando com as palhetas dos restantes modelos que são polidas (figura 1.3).

Nas figuras 1.1 e 1.2 podem-se visualizar alguns dos componentes de um dos modelos (R9M) da bomba de óleo de caudal variável:

- | | |
|-------------------|-----------------------|
| 1. Corpo; | 5. Palhetas do rotor; |
| 2. Estator; | 6. Pinhão; |
| 3. Rotor; | 7. Segmento; |
| 4. Veio do rotor; | 8. Mola. |

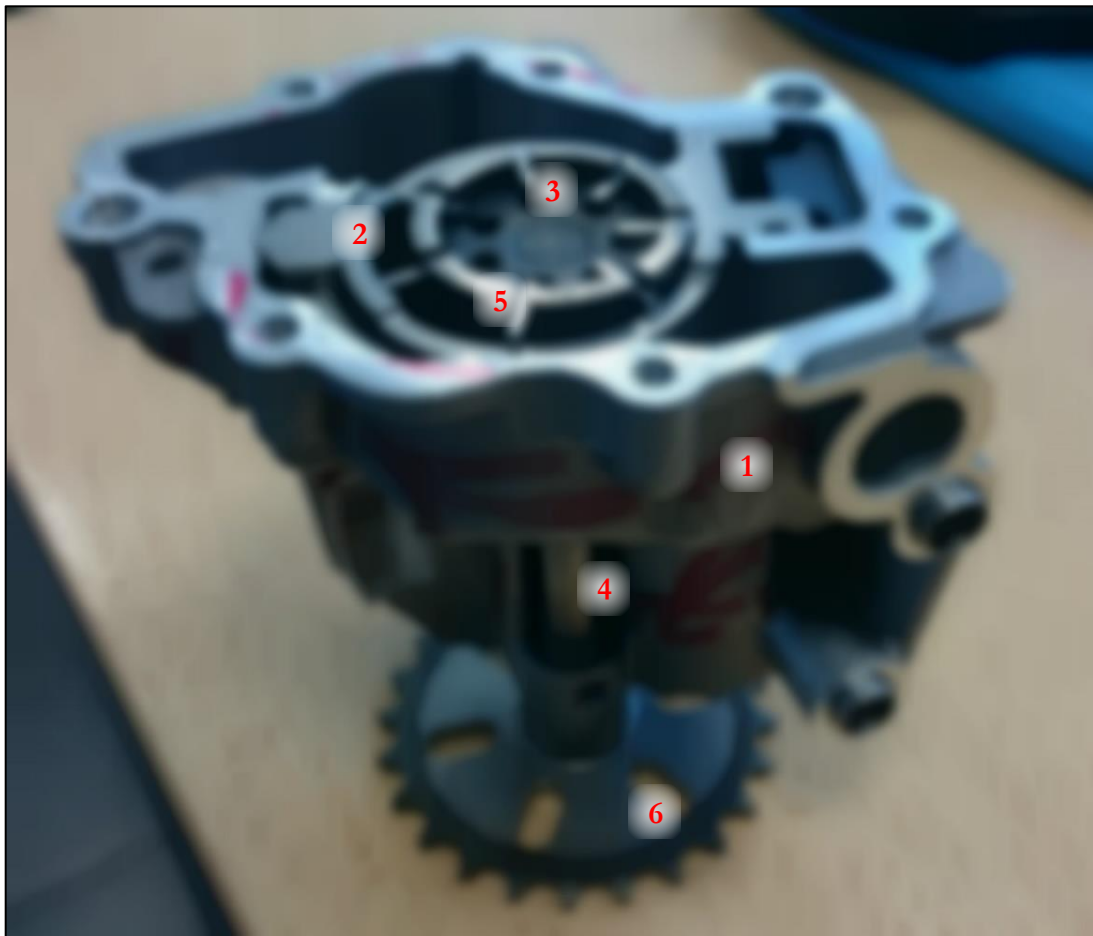


Figura 1.1: Bomba de óleo de caudal variável (R9M, montagem parcial), vista geral.

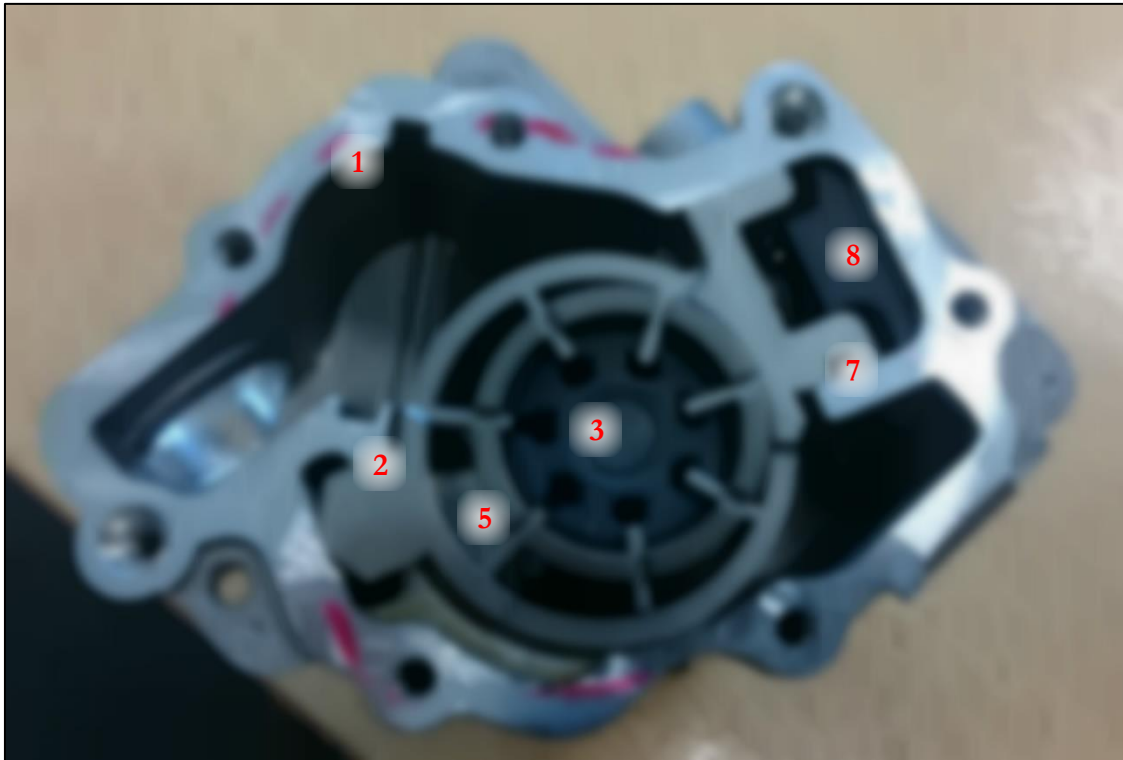


Figura 1.2: Bomba de óleo de caudal variável (R9M, montagem parcial), vista de cima.

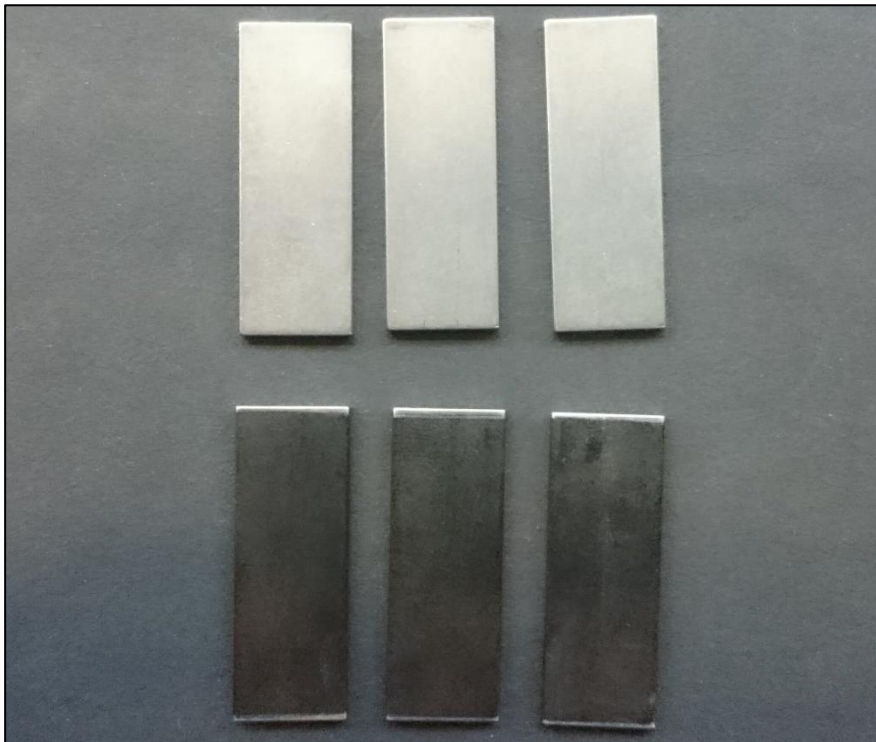


Figura 1.3: Palhetas polidas e escuras.

1.3. Linha de Montagem de Bombas de Óleo de Caudal Variável

A linha de montagem de bombas de óleo de caudal variável tem como objectivo ser uma solução “chave na mão”, ou seja, pronta a utilizar. A formação aos operadores da linha e documentação necessária são também fornecidas. Este sistema é quase totalmente automatizado, tendo apenas cinco locais em que existe manuseamento manual, sendo este manuseamento limitado à colocação de componentes nos locais apropriados do produto (montagem manual) e transporte, ou abastecimento da linha. A linha de montagem é um sistema flexível, permitindo a montagem dos vários modelos de bombas de óleo, mencionados anteriormente, através da variação na sequência e tipo de movimentos do sistema, permitindo criar diferenças posicionais, dimensionais e de componentes das bombas.

1.3.1. Postos de Trabalho

Este sistema é formado por vários postos de trabalho, sendo que cada um deles realiza uma ou mais tarefas específicas na montagem das bombas de óleo. Estes postos de trabalho são denominados “OP” (“operação”), existindo também um banco de ensaios – cuja responsabilidade não é da Atena ou estagiário – destinado a testes de funcionamento das bombas de óleo. As operações individuais nos postos são identificadas como valores decimais dos OP, como por exemplo, OP70.1, ou OP90.3. Postos de trabalho com numeração intermédia, inferior ou superior não fazem parte do âmbito deste projecto.

Por ordem de montagem, apresentam-se os postos de trabalho (identificados na figura 1.4), com breves descrições das suas operações:

1. OP15: montagem de um filtro e uma válvula no corpo da BOCV através de prensagem. Manipulação humana para posicionamento dos componentes. Apenas no modelo Hxx;
2. OP25: montagem de dois casquilhos e uma válvula no corpo da bomba, utilizando prensagens para ambas as operações. A montagem da válvula tem manipulação humana para colocação dos componentes a montar e passagem do produto entre este posto e o de controlo do pinhão;
3. OP40: montagem de filtro num componente cilíndrico, resultando num componente constituinte da válvula montada no posto OP25. Este componente, um pistão, destina-se apenas aos modelos M9T e R9M;

4. Posto de controlo do pinhão (sem designação OP): verifica a existência do pinhão da bomba de óleo na palete;
5. OP45: montagem do estator no corpo da bomba de óleo e montagem da mola que permite a variabilidade na geometria necessária para o caudal poder ser variável. O posto tem manipulação humana para posicionamento dos componentes, com posterior montagem automática;
6. OP60: lubrificação do veio do rotor da bomba e montagem no núcleo. Um braço robótico é utilizado para manipulação e posicionamento das duas peças e colocação da montagem final no posto seguinte. É utilizada uma prensa para efectuar a montagem do veio no rotor;
7. OP70 (posto de lubrificação): lubrificação da montagem parcial;
8. OP70 (posto de introdução de anilhas): introdução de anilhas no sistema para posterior montagem no corpo da bomba de óleo (postos OP70.1 e OP70.3);
9. Posto de introdução de palhetas: introdução de palhetas no rotor da bomba de óleo;
10. OP70.1: montagem de anilha no corpo da bomba de óleo;
11. OP70.2: montagem do rotor com núcleo, veio e palhetas no conjunto corpo e estator, através de uma prensa. Também é realizada a montagem do pinhão no veio. A existência de um braço robótico permite o transporte do conjunto do rotor do posto de introdução de palhetas para este posto;
12. OP70.3: montagem de anilha no corpo do produto;
13. OP70.4: aparafusamento de electroválvula no corpo da bomba de óleo (apenas para o modelo Hxx);
14. OP70.5: sistema de visão de verificação da correcta montagem de palhetas e anilhas no conjunto (com análise extra de segmento);
15. OP70.6: posto de lubrificação;
16. OP90.1: montagem da tampa no corpo da bomba de óleo, seguida de aparafusamento. Braço robótico realiza a tarefa de posicionamento da tampa no corpo;
17. OP90.2 e OP90.3: aparafusamento dos parafusos em falta no produto. Cada posto de trabalho é constituído por um robô que realiza a tarefa;
18. Banco de ensaios: realiza testes de bom funcionamento das bombas de óleo;
19. OP160: marcação laser de *data matrix* na bomba de óleo, para futura identificação e rastreabilidade.

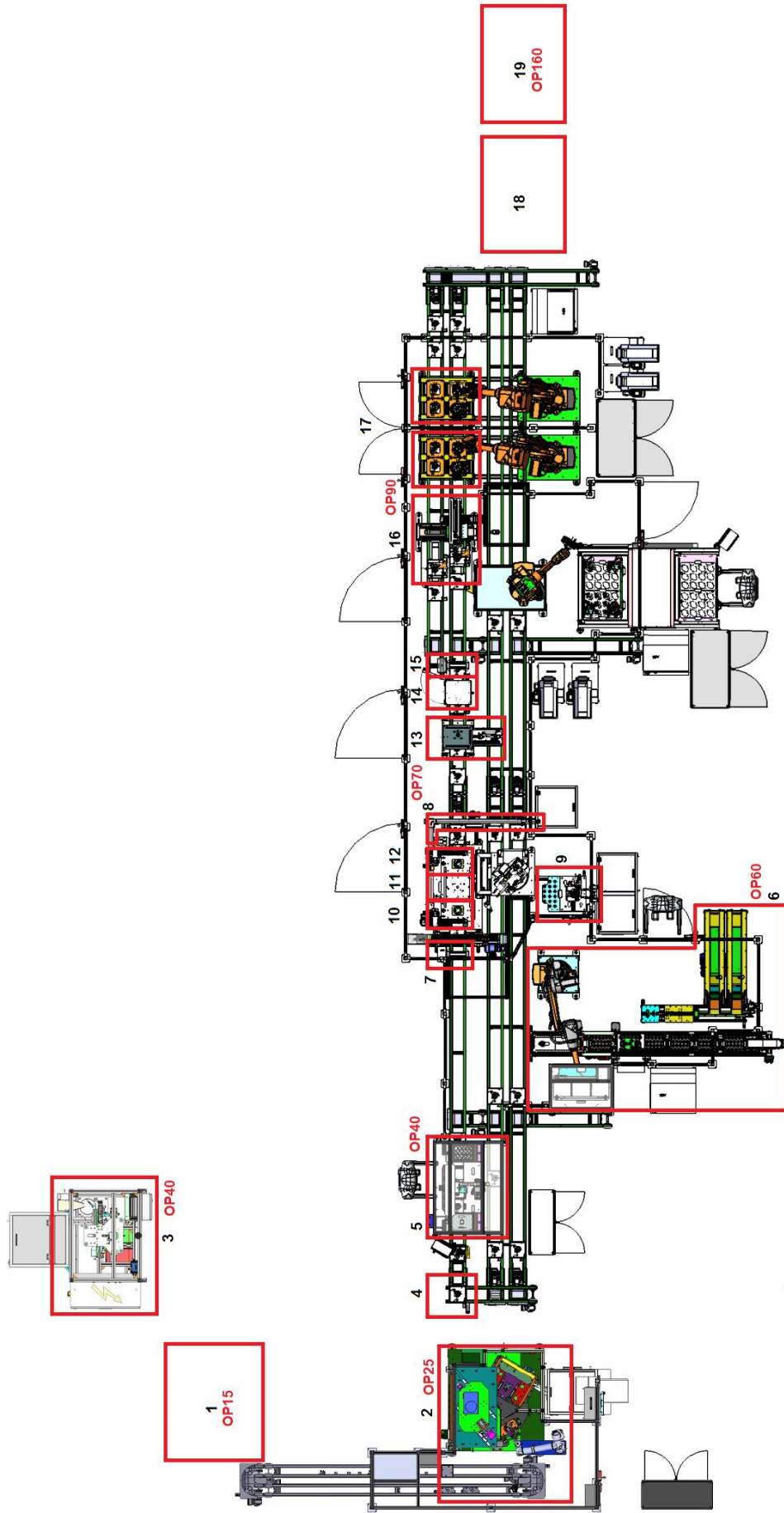


Figura 1.4. Linha de montagem de bombas de óleo de caudal variável: esquema geral, com postos de trabalho identificados.

1.3.2. Transportadores

Além dos postos de trabalho, existem também os transportadores dos componentes e produtos entre postos de trabalho. Estes sistemas, localizados entre o OP15 e OP25 e entre o posto de detecção de pinhão e o OP90 são constituídos por tapetes rolantes, dotados de locais de paragem, controlados pelas zonas de trabalho, quando necessário – por exemplo para espera antes da realização de operações em cada posto. Para accionamento são utilizados uma série de motores eléctricos. O produto montado parcialmente é transportado em paletes. Existem também sensores fotoeléctricos e indutivos para detecção das bombas de óleo e paletes, em locais específicos, como por exemplo antes dos locais de espera, ou para detecção antes de serem efectuadas operações de montagem nos postos de trabalho.

É empregue um sistema de retorno para transportar paletes vazias no fim de ciclo dos sistemas para o início, de modo a poderem ser reutilizadas para transportar novas bombas de óleo em processo de montagem.

Tal como as zonas de postos de trabalho, os sistemas de transporte têm um sistema de controlo implementado através de autómatos programáveis. O transportador entre o OP15 e a OP25 é controlado directamente pelo autómato do último posto, enquanto que o segundo sistema transportador possui um autómato dedicado.

As paletes que viajam ao longo dos transportadores (figura 1.5) são constituídas por uma base, uma *tag* RFID (apresentada mais adiante), suportes da bomba de óleo e furações para poderem ser bloqueadas por cilindros ao longo dos transportadores. Existem três modelos de paletes: o primeiro, mais pequeno que os restantes, transporta os corpos das bombas de óleo (dos três modelos) entre o posto OP15 e o OP25. Os restantes dois tipos servem para transportar as bombas de óleo entre o posto de controlo de pinhão e o OP90, com um deles transportando bombas Hxx e o outro M9T e R9M. Além de transportarem bombas em processo de montagem, os dois últimos tipos de paletes podem ainda transportar componentes a montar, como pinhões. Na figura 1.5 encontra-se apresentada uma paletes do segundo tipo, para transporte de bombas Hxx. Pode-se observar os três postes de suporte da bomba, assim como no centro um local de colocação do pinhão antes da sua montagem e, no canto superior direito um suporte para a *tag* RFID.

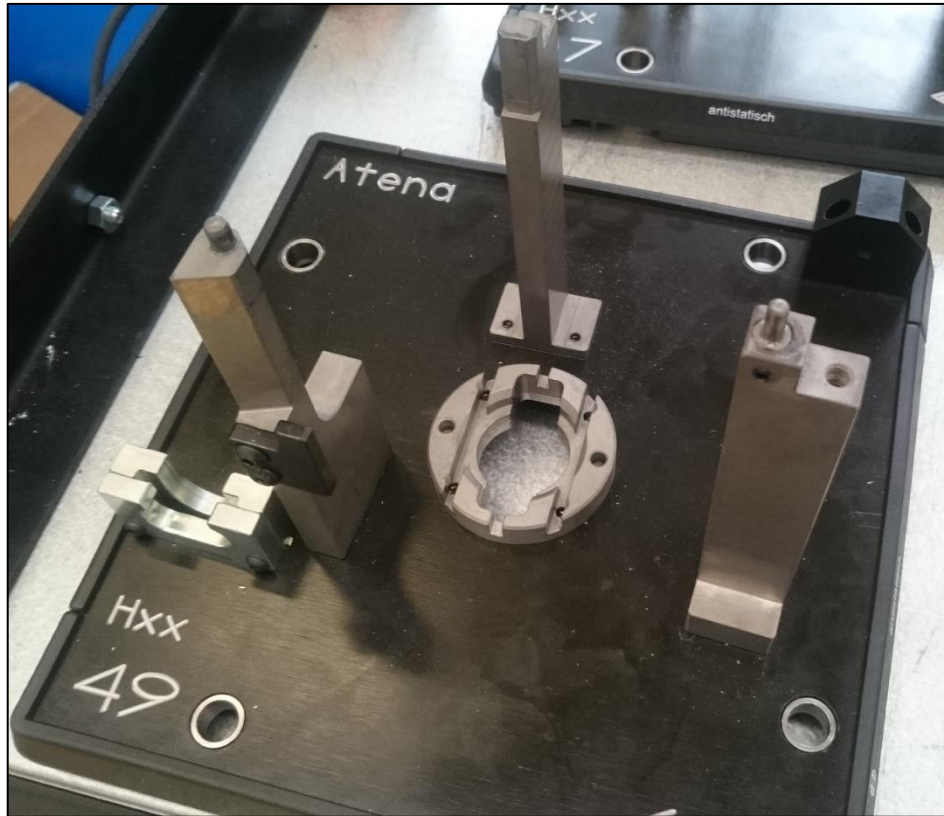


Figura 1.5: Palete de bombas Hxx entre o controlo de pinhão e a OP90.

1.3.3. Sistemas de Identificação RFID e Leitura de Códigos de Barras

A linha de montagem é dotada de um sistema de identificação das paletes por RFID (consultar a obra de Finkelzeller [5] para mais detalhes) – *Radio-Frequency Identification* – utilizando *tags* passivas. Este sistema é utilizado para informar os postos de trabalho das características da bomba de óleo a montar, incluindo modelo, informações dimensionais desta e outras informações necessárias, tais como a força a ser aplicada por prensas, ou o binário a utilizar por aparafusadoras. Também transmite informação acerca do estado de montagem destas. Este sistema baseia-se na existência de *tags* nas paletes que serão lidas por leitores ao longo da linha. Estas *tags* podem também ser escritas, sendo actualizadas após cada operação em cada posto de trabalho. Estas *tags* dinamizam a tarefa da paleta, tornando-a um verdadeiro objecto inteligente [6]: o fluxo de informação permitido por estas entre postos de trabalho permite uma grande flexibilidade na produção, oferecendo a possibilidade de realizar lotes de pequenas dimensões (ou mesmo unitários) sem grandes perdas de rendimento (com estas a acontecer apenas na troca de ferramentas nos postos com manipulação manual).

Para transmissão de dados entre postos de trabalho não servidos por transportadores são utilizados leitores de códigos de barra para leitura da etiqueta existente nos corpos de cada

bomba individual, que são lidos antes da introdução destas nos postos. Estes sistemas de identificação possibilitam rastreabilidade e controlo de qualidade, com qualquer defeito de fabrico identificado e registado. Dependendo desta informação o sistema rejeita ou não a bomba de óleo a trabalhar ou após ser trabalhada por um posto de trabalho, não sendo desperdiçados ciclos das máquinas e componentes em bombas defeituosas parcialmente montadas.

1.3.4. Sistemas de Controlo

O controlo e accionamento da linha de montagem é implementado por vários autómatos (PLC – *Programmable Logic Controller* [7]), um por posto de trabalho, excluindo o OP45 e OP60 cujo controlo é efectuado por um único sistema. Os autómatos utilizados fazem a leitura da informação fornecida pelos vários sensores, relacionam esses dados com os retirados de um sistema de gestão e activam ou desactivam variados componentes integrantes do sistema, como válvulas, motores e outros. As zonas de trabalho e, conseqüentemente, os sistemas de controlo destas, agem de modo independente entre si (com excepção da OP45 e OP60, explicado anteriormente), sendo a única troca de informação realizada pela escrita ou leitura das *tags* RFID e leitura dos códigos de barras.

Outro componente do sistema de controlo são as interfaces de comunicação homem-máquina – HMI [7]. Estas interfaces têm o objectivo de oferecer aos operadores uma fonte de informação rápida e fácil do funcionamento do sistema, e permitir o funcionamento manual, servindo as interfaces como comandos para controlo directo de todos os componentes manipuláveis da linha. Também são utilizadas para apresentar alarmes, avisando os operadores de algum problema que possa estar a acontecer na linha.

1.3.5. Sistema de Gestão de Produção

O sistema de gestão de produção mencionado anteriormente serve para comunicar à linha a informação de fabrico (ordens de produção e configuração), lendo o estado da produção em cada um dos postos de trabalho.

Este sistema, do tipo SCADA [8] – *Supervisory Control and Data Acquisition*, baseia-se num servidor. O servidor, através de receitas de produção contidas numa base de dados SQL, obtém as ordens de produção. Estas ordens de produção (receita) são escritas nas *tags* RFID das paletes e associadas aos códigos de barras de cada bomba de óleo de forma integral, permitindo assim aos vários postos de trabalho saber qual o modelo de bomba de óleo a montar. Por outro lado, o sistema recebe a informação da linha de montagem quanto aos resultados obtidos – número

de unidades produzidas, quantidade de bombas com defeito, operações de montagem mal realizadas e outros parâmetros de montagem como binários de aperto ou cotas de prensagem obtidos – obtendo-se assim um número de estatísticas e históricos de produção que servirão para efeitos de supervisão e registo. Estes resultados são armazenados também eles numa base de dados SQL (diferente da das receitas), o que obriga o sistema de gestão a ser capaz de inserir, consultar e apagar dados em tabelas destas. É também este sistema que gere alguns alarmes e avisos gerados pela linha de montagem, além de possuir um ambiente gráfico para fácil visualização de todos os dados necessários.

1.3.6. Sistemas de Segurança Máquina

Vários sistemas de segurança máquina estão associados à linha de montagem, para proteger os operadores de possíveis acidentes. Este é um dos sistemas mais importantes, se não o mais importante, já que a protecção de vida humana se sobrepõe à máquina e à sua produção. Os componentes essenciais são relés de segurança, que realizam paragens do equipamento seguras, se certas condições não forem cumpridas: activação da paragem de emergência ou qualquer outro dispositivos de segurança ligado a estes relés. Exemplos desses dispositivos na linha de montagem são barreiras imateriais fotoeléctricas, botões de emergência e fins de curso de segurança de portas fechadas. Além disso existem também barreiras físicas: painéis de policarbonato, barreiras de protecção de rede, chaves para trancar quadros eléctricos e comandos bimanuais no caso de interacção directa com o operador (exemplo da OP45).

1.3.7. Prensas e Aparafusadoras

Para efectuar algumas operações de montagem utilizam-se prensas e aparafusadoras. Estas necessitam de um controlo preciso da força a aplicar nos componentes a montar e da cota de introdução de um componente, no caso da prensagem, e do binário de aperto, no caso do aparafusamento. A incorrecta aplicação de força por uma prensa pode causar a montagem incorrecta, se não for suficiente, e possivelmente deformar as peças, se for excessiva. Quanto ao binário de aperto, se este for baixo não garantirá o aperto correcto dos parafusos, podendo estes soltar-se durante o funcionamento da bomba de óleo (devido a vibrações, por exemplo). Se for demasiado elevado, poderá deformar o parafuso, ou até mesmo causando a sua ruptura, impedindo o seu desaperto e consequentemente a manutenção da bomba, ou levando à falha total desta.

1.3.8. Descrição de Conjunto

Resumidamente, a linha de montagem de bombas de óleo de caudal variável é um conjunto de postos de trabalho automatizados que possibilitam a montagem de vários modelos de bombas de óleo com ligeiras diferenças entre si, sendo que esta produção é controlada automaticamente através de sistemas de controlo e gestão, e manualmente, caso necessário, utilizando-se interfaces homem-máquina, que também permitem a visualização de dados de funcionamento e produção. A automatização inclui não só a produção em si, mas também o controlo de qualidade – utilizando, por exemplo, *Poka-Yokes* [9] – e as medidas de segurança implementadas, não apenas para a máquina, mas, mais importantemente, para o utilizador.

Este projecto insere-se no conceito de “Indústria 4.0” [10] [11] com a utilização maciça de dispositivos conectados em rede e troca de informação entre cada elemento da linha de montagem. Isto permite não só o armazenamento de informação relativa à produção de um modo fácil e flexível, como também o processamento dessa informação para despiste de problemas com o sistema e identificação de áreas a melhorar, com a finalidade de ser possível um aumento da qualidade do produto final.

Um diagrama de blocos das interações dos sistemas integrantes da linha de montagem pode ser visualizado na figura 1.6. A figura 1.7 representa a organização geral de um posto de trabalho genérico.

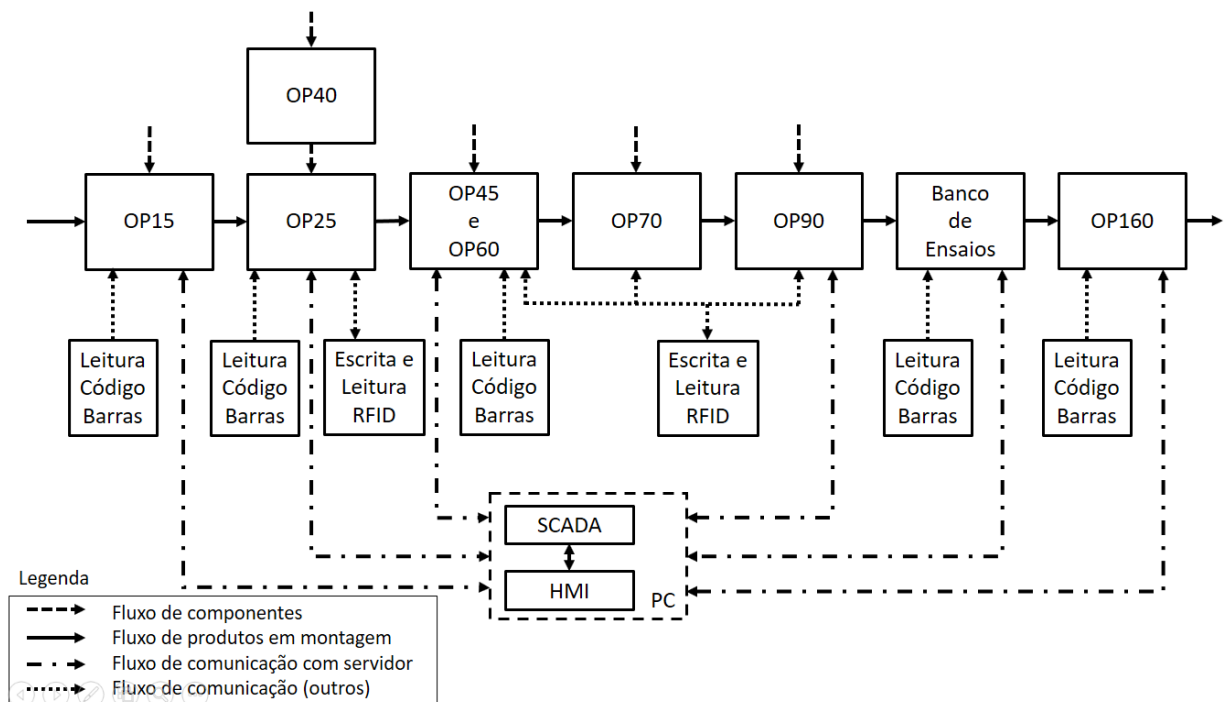


Figura 1.6: Diagrama de blocos da linha de montagem de bombas de óleo.

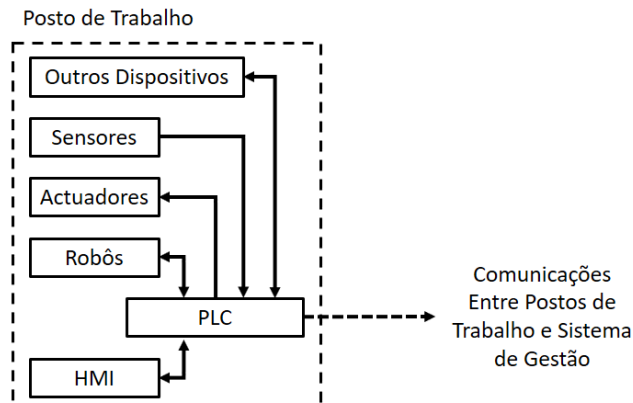


Figura 1.7: Diagrama de blocos de um posto de trabalho genérico.

1.4. Objectivos do Estágio

Após a introdução ao problema resta expor os objectivos do estágio, de forma mais concreta. Tendo os esquemas de princípio (esquematisação do sistema), mecânico e pneumático da linha de montagem apresentada finalizados, é necessário realizar os projectos eléctrico e de automação.

O projecto eléctrico envolve a criação dos esquemas eléctricos do sistema, incluindo todos os componentes eléctricos, electrónicos e electromecânicos, assim como as ligações entre estes. Já o projecto de automação, mais abrangente, inclui a necessidade de desenvolver a programação dos autómatos, interfaces homem-máquina e sistema de gestão de produção. Terá que ser efectuada também a programação de manipuladores robóticos, assim como desenvolver-se o sistema de visão a empregar no posto de trabalho OP70.5, e criar sistemas de comunicação entre todos os componentes e dispositivos necessários, através da utilização de protocolos de comunicação de redes industriais.

Este projecto implica o desenvolvimento em conjunto pela equipa de todos os sistemas da linha de montagem, sendo que cada um poderá ser atribuído a uma única pessoa, ou ser trabalhado por mais que um elemento. No que respeita às componentes de trabalho que foram desenvolvidas de forma individual, coube ao candidato a assistência aos restantes elementos da equipa e parceria com estes no desenvolvimento dos sistemas, de forma geral, e desenvolvimento individual de algumas partes da linha de montagem, nomeadamente o desenvolvimento conjunto com mais um colega de trabalho do servidor de gestão de produção, desenvolvimento

do sistema de visão utilizado no posto OP70.5, desenvolvimento em conjunto com outros elementos da equipa do posto OP25 e criação e/ou actualização de esquemas eléctricos de diversas partes do projecto e criação de interfaces homem-máquina.

Outro objectivo é a familiarização com *hardware* e *software* utilizados na indústria, desde tipos de autómatos, sensores, actuadores, sistemas de segurança, até programas e ecossistemas de desenvolvimento.

Por último, será realizada uma análise dos programas desenvolvidos procurando padrões – trechos de código, passíveis de reaproveitamento – e anti-padrões – código confuso, convoluto ou ineficiente – na tentativa de criar ferramentas de ajuda a uma maior eficiência e rapidez de programação.

A actividade foi repartida entre dois locais: a empresa e o cliente para a qual se destina a linha de montagem, por motivos de implementação.

O capítulo 4 – Sistemas de Visão – e seguintes apresentam todo o trabalho desenvolvido pelo estagiário, excepto quando explicitamente mencionado.

2. ESTADO DA ARTE

2.1. Introdução

O presente capítulo tem o objectivo de apresentar as tecnologias utilizadas ao longo do estágio e suas considerações técnicas e científicas, incluindo uma apresentação do projecto eléctrico e de automação.

2.2. Projecto Eléctrico

O projecto eléctrico é efectuado após a realização do projecto mecânico e pneumático, ou em paralelo com estes, e antecede o projecto de automação. Deste resulta toda a informação necessária para interligar todos os actuadores, sensores e outros componentes que necessitam de electrificação e/ou controlo, assim como os dispositivos a utilizar para atingir esse fim. Com o projecto eléctrico concluído, é possível realizar a construção do quadro eléctrico do sistema e a ligação deste à rede eléctrica, e a conexão do sistema aos sensores, actuadores e outros componentes (HMI, RFID e outros).

2.2.1. Fases do Projecto Eléctrico

No projecto eléctrico é necessário identificar e seleccionar os componentes a utilizar: dispositivos de segurança e protecção eléctricos, comandos e accionadores de motores, contactores, relés auxiliares, dispositivos de controlo (autómatos e dispositivos associados), componentes pneumáticos, e outros. O passo seguinte é a análise de risco e segurança do equipamento. Daqui surge a selecção de componentes de segurança humana em função dos seus níveis de desempenho, como interruptores diferenciais, componentes de accionamento de emergência –

botões de emergência, barreiras imateriais, entre outros – e bloqueios de acesso, como fechos magnéticos. Procede-se então para o cálculo das necessidades energéticas do sistema, que vão resultar na escolha da forma de fornecimento de energia à instalação – número de fases e corrente a utilizar – e equipamentos de corte e protecção e transformação de energia (como fontes de alimentação).

Tendo sido feita a selecção completa de componentes de controlo e potência, torna-se necessário transferir essa informação para o esquema eléctrico, que contém também todas as ligações entre componentes de controlo e potência. Finaliza-se o projecto eléctrico efectuando-se uma análise funcional para a construção do equipamento, ou seja, de que modo é mais simples e intuitiva a construção do quadro eléctrico e disposição da cablagem [12].

2.2.2. Esquema Eléctrico

Um esquema eléctrico é um esquema ou diagrama que mostra de que modo é que os vários componentes de um circuito eléctrico vão estar interligados entre si. Estes esquemas podem ter vários métodos de organização dos componentes e ligações (com alguns exemplos apresentados no Anexo 1):

- Organização por referência: apresenta uma vista detalhada de todos os elementos do quadro eléctrico e suas referências, sendo por isso bom para a construção do equipamento. Empresas alemãs recorrem habitualmente a este tipo de organização;
- Organização por função: agrupa todos os componentes que executam uma função numa única página – por exemplo, apresenta o circuito de controlo e potência de um motor específico numa página. É bom para procura e solução de problemas eléctricos, caso surjam, além de simplificar o desenho, já que o projectista não necessita guiar-se por referências que possam estar noutras páginas, mas apresenta descontinuidade na organização de entradas e saídas de comando e potência, o que dificulta saber quais as que estão em utilização e quais estão de reserva;
- Organização por componente: visão global simplificada, quando comparado com a organização por função, sendo por isso mais fácil de consultar, mas mais difícil para montar o equipamento. A organização costuma ser por tipo de componente – todos os motores utilizados estão agrupados numa página, como por exemplo cartas de entrada dos autómatos todas agrupadas. É utilizado habitualmente por empresas francesas.

A organização das ligações entre componentes, as regras de organização de um esquema eléctrico – desde como realizar uma lista de componentes, à simbologia utilizada para descrever

componentes e cabos, assim como a sua referenciação, até ao tipo de cabos a utilizar – estão compiladas numa série de normas do CENELEC – *European Committee for Electrotechnical Standardization*, associadas ao IEC – *International Electrical Commission* (no caso de Portugal [13] [14] [15]). Estas incluem:

- IEC 60617: simbologia gráfica para diagramas [16] [17];
- IEC 61175 e IEC 81346: designação, estruturação e referenciação de sinais e conexões [18] [19];
- IEC 62023: estruturação de informação técnica e documentação [20];
- IEC 62027: organização de listas de componentes [21];
- IEC 60445: princípios básicos de interface homem-máquina, incluindo identificação de equipamentos, cablagem e conectores [22];
- IEC 60204: designações de referência de equipamentos industriais [23];

2.3. Projecto de Automação

O projecto de automação envolve, como o nome indica, o desenvolvimento da automação de processos e acções efectuadas por uma máquina. Actualmente, recorre-se a autómatos programáveis para criar sistemas automáticos de controlo robustos e flexíveis, ao mesmo tempo eliminando a necessidade de ligações eléctricas fixas associadas aos métodos tradicionais de controlo por lógica de relés físicos [24]. Todos os componentes de actuação e sensores agem como periféricos do PLC, que lê dados, processa-os e, consoante o resultado, activa ou não componentes do sistema. Para o autómato comunicar com o utilizador, e vice-versa, recorre-se à sinalética – como sinais visuais ou sonoros – e formas de introdução de comandos, como botões. Métodos de comunicação mais avançadas consistem em HMI (*Human-Machine Interface*), capazes tanto de oferecer visualização de dados como de aceitar comandos, que transmitem ao autómato. Um último subsistema que pode ser associado ao PLC é o sistema de gestão de produção.

Relativamente à organização dos dispositivos no sistema, estes seguem uma hierarquia bem definida de controlo. No nível mais baixo encontram-se os sensores, actuadores, e outros dispositivos de entrada e saída. Acima destes está o PLC, que os controla. No nível seguinte encontra-se o sistema de supervisão e gestão (incluindo sistemas HMI e SCADA), com o servidor da unidade fabril no nível seguinte, constituindo um sistema MES (*Manufacturing Execution System* – gestão e execução de produção) [25]. Por último, surge o sistema de planeamento de recursos de uma empresa – ERP, *Enterprise Resource Planning*. Pode-se observar essa hierarquia na figura 2.1.

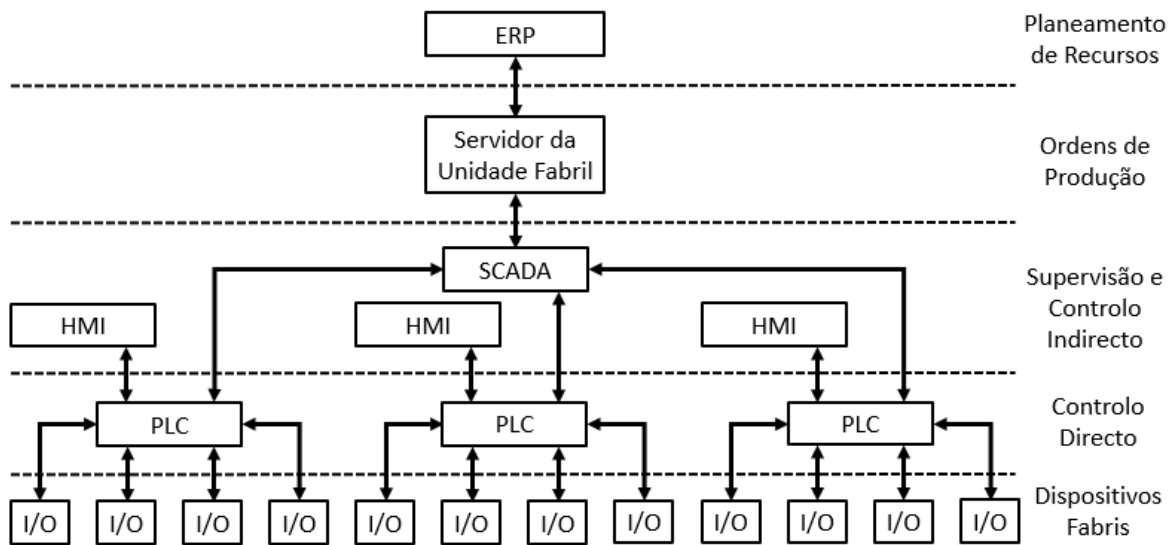


Figura 2.1: Hierarquia de controlo de um sistema fabril.

É necessário no projecto de automação a integração de vários dispositivos, utilizando para isso variadas tecnologias, existindo um grande número de formas de abordar o mesmo problema. Essa flexibilidade torna-se um encargo cada vez maior quanto mais complexo é o sistema a desenvolver, pois tem que se encontrar o melhor método de lidar com um grande número de periféricos e uma enorme quantidade de dados a serem transmitidos e processados, tendo origens e destinos diferentes, dependendo da organização do problema [26]. É necessário, portanto, bastante pesquisa e teste de todas as possibilidades, simulando ou experimentando com dispositivos físicos. Essa pesquisa e desenvolvimento tem também que contemplar a obrigatoriedade de criação de sistemas que não falhem – extremamente improvável – ou que, apesar de raramente falharem, contenham medidas para lidarem contra essas falhas, de maneira a não permitir a falha total geral. O objectivo do projecto de automação é, não apenas desenvolver um produto que funcione, mas também que funcione bem, tendo em conta as necessidades finais para qual o sistema automático se destina.

2.3.1. Autómatos Programáveis

Um autómato programável é um computador desenvolvido para ser usado em sistemas de controlo industrial. Devido à utilização na indústria, os PLC's são desenhados para serem robustos, tanto fisicamente como operacionalmente, ou seja, estão preparados para uma operação contínua durante milhares de horas em condições ambientais muito extremas a nível de temperaturas, humidade, vibrações, partículas, entre outros. Estes dispositivos são formados por entradas e saídas de sinais digitais e analógicos e uma unidade central de processamento (CPU), que realiza a interpretação de sinais recebidos nas entradas e dos programas armazenados na memória, comunicando as suas decisões às saídas. É possível que certos modelos de

autómatos possuam outros componentes especializados, tendo-se o exemplo de cartas de comunicação (dos mais variados protocolos de comunicação), controladores de motores e outros. Outra característica dos autómatos é a sua capacidade de comunicar com outros dispositivos, nomeadamente outros autómatos e computadores pessoais, através de uma variedade de interfaces de comunicação. Algumas unidades também possuem fontes de alimentação (PSU) incorporadas [25]. Um esquema dos blocos constituintes de um PLC pode ser visualizado na figura 2.2.

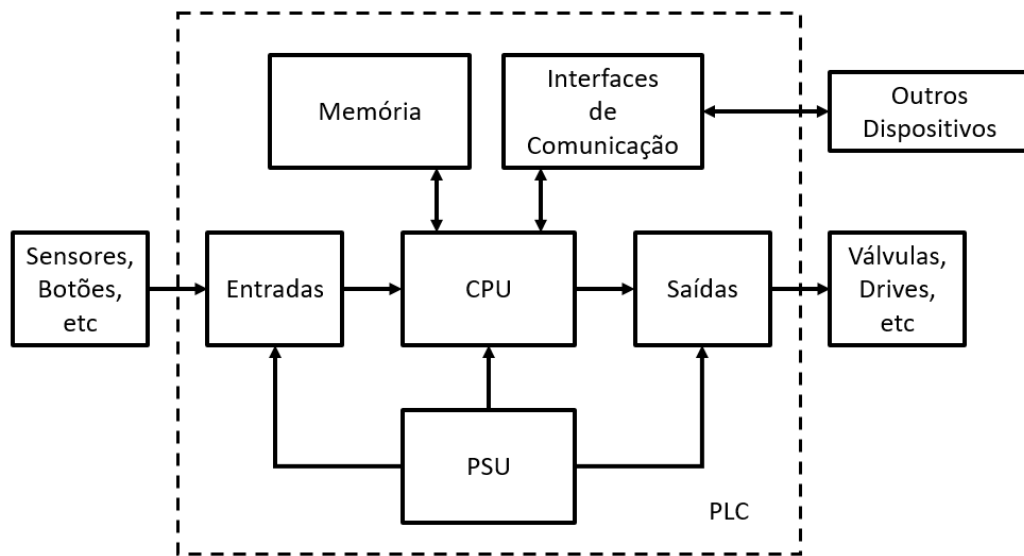


Figura 2.2: Esquemática de um PLC e dispositivos periféricos.

A programação destes dispositivos é feita através de um computador externo, com *software* específico de cada fabricante, sendo o programa desenvolvido e depois transferido para a memória do autómato e executado automaticamente. É possível também exportar programas residentes na memória para um computador externo, o que oferece bastante flexibilidade durante o desenvolvimento e suporte de sistemas de controlo [24]. Existem várias possibilidades quanto à linguagem de programação de um PLC, sendo as linguagens standard a todos os fabricantes de autómatos definidas na norma IEC1131-3 [27]. Uma é a lógica *Ladder* (LAD), uma linguagem gráfica de diagramas. Foi criada com o intuito de oferecer uma baixa curva de aprendizagem para engenheiros e técnicos que dominavam a lógica e construção de sistemas de controlo baseados em lógica de relés, já que esta linguagem baseia-se nesta lógica. Para sistemas relativamente simples é um método de programação bastante directo. Outra linguagem utilizada designa-se por FBD – *Functional Block Diagrams*. Esta também é uma linguagem gráfica e baseia-se em blocos interligados de funções, oferecendo possibilidades de processamento de sinais complexos ou permitindo a simplificação de programas *Ladder* em algumas situações. Relativamente a linguagens não-gráficas, existem a *Instruction List* (IL), uma linguagem de programação de baixo nível com parencas com *assembly*, boa para aplicações que exigem um controlo muito

rápido (uma comparação entre estas três linguagens encontra-se na figura 2.3), e *Structured Text* (ST), uma linguagem de alto nível semelhante ao Pascal (figura 2.4). Uma última linguagem, *Sequential Function Chart* (SFC), toma a forma de diagramas de blocos e é destinada a aplicações que necessitem de um grande paralelismo, sendo que várias sequências de controlo podem ou não ser activadas simultaneamente (figura 2.5). Esta última foi baseada em GRAFCET [28] [29]. Os autómatos programáveis são soluções proprietárias, o que significa que, apesar de existirem linguagens de programação estandardizadas, não significa que os fabricantes dos dispositivos as implementem na íntegra, podendo estar adaptadas, ou que incluam a possibilidade de programação utilizando todos os métodos acima descritos.

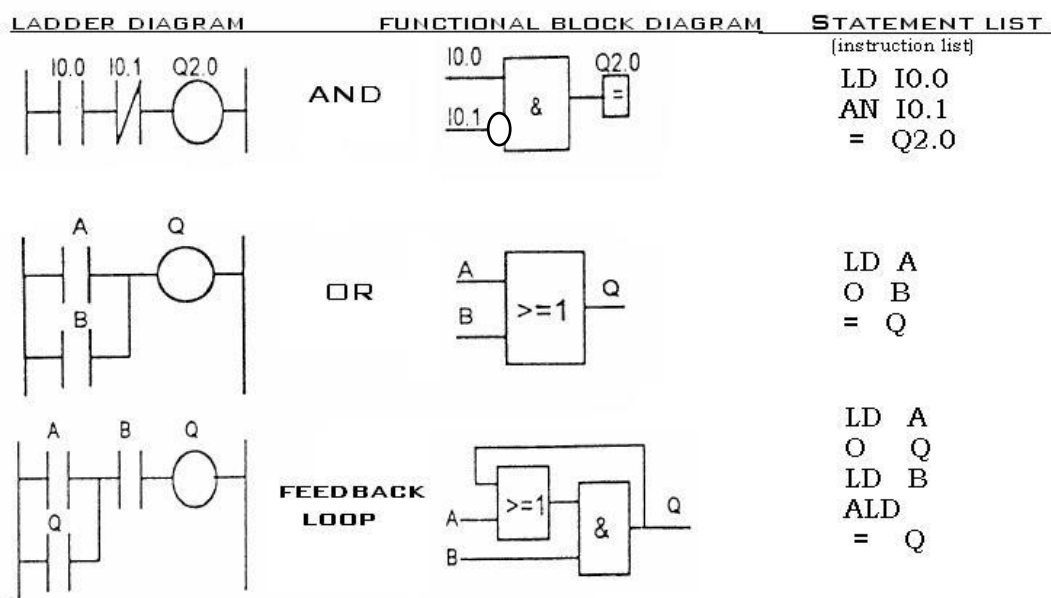


Figura 2.3: Comparação entre LAD, FBD e IL, utilizando um “AND” e um “OR” lógicos e um circuito de retorno [30].


```

Program Main
  Var T : TagWriteValue; End_Var
  Var DoReq : Bool; Index : INT; End_Var

  if Index < 11 Then

    if T.DONE Then
      (* Move to the next request *)
      Index := Index + 1;
      DoReq := False;
    else
      (* Start the next request *)
      DoReq := True;
    end_if;

    T( REQ:=DoReq,
      TAGNAME:=TagNames[ Index ],
      Value:=TagValues[ Index ] );
  end_if

end_Program

```

Figura 2.4: Exemplo de programa em ST [31].

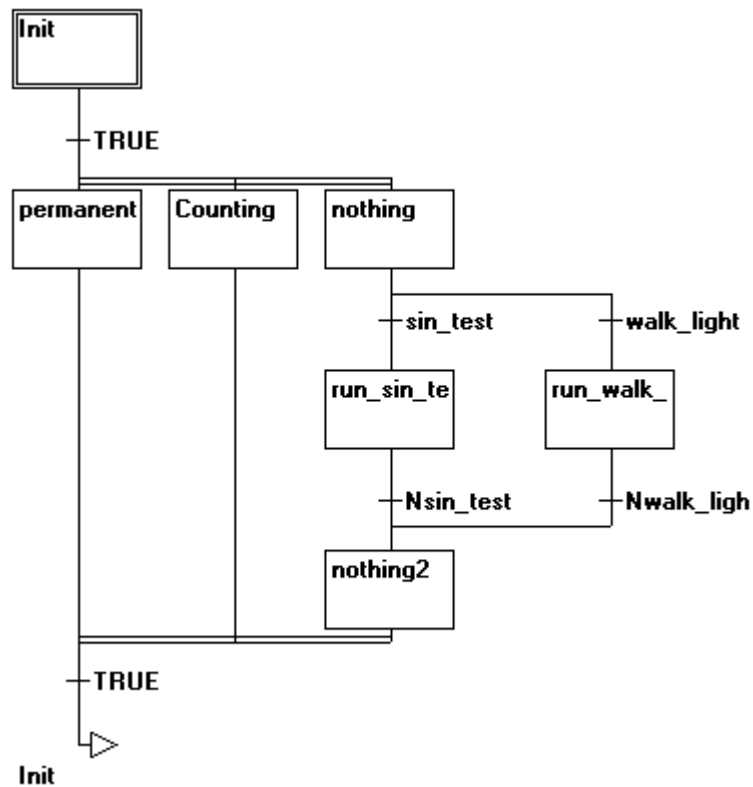


Figura 2.5: Exemplo de programa utilizando SFC [32].

2.3.2. Interfaces de Comunicação

Actualmente existe uma miríade de interfaces de comunicação entre sistemas de controlo e outros dispositivos [33] (sendo a família de interfaces *fieldbus* [34] um exemplo), com os fabricantes de equipamento geralmente suportando apenas um número reduzido de protocolos nos seus dispositivos. Regra geral, estas interfaces simplificam e flexibilizam a comunicação, muitas vezes eliminando a necessidade de utilizar entradas e saídas dos autómatos – o que reduz o trabalho de cablagem e a quantidade desta utilizada – e permitindo uma maior velocidade de transmissão de dados entre estes e dispositivos associados. Estes métodos de comunicação são baseados em tecnologias de comunicação série, e são optimizadas para transmissões rápidas de pacotes de dados – as mensagens enviadas através de interfaces industriais, regra geral, limitam-se a poucos bytes de tamanho – de modo a reduzir ao máximo o *overhead* existente em outras tecnologias, como o TCP/IP [35].

A ligação física é habitualmente feita de cabos, mas a utilização de tecnologias sem fios tem vindo a ter uma implementação limitada na indústria, com o objectivo de flexibilizar ainda mais a ligação entre dispositivos e sistemas. Estas tecnologias têm o potencial de baixar custos de implementação de equipamentos e aumentar a produtividade – por exemplo, permitindo a existência de interfaces homem-máquina portáteis – mas sofrem de problemas que dificultam a sua aplicação generalizada: possibilidade de falhas de segurança, e existência de interferências de radiofrequência e outras, sendo por isso ainda pouco utilizadas a nível industrial [36] [37], apesar de existirem já métodos para combater algumas das suas desvantagens [38].

2.3.3. Interfaces Homem-máquina

As interfaces entre máquinas e humanos podem variar de complexidade, desde simples botões e sinais luminosos ou sonoros, até inserção de eléctrodos no corpo humano para controlar próteses robóticas. No caso da indústria, está completamente difundida a utilização de botões de comando, geralmente para activar e desactivar sistemas, sinais luminosos, como sistemas de sinalização *Andon* [39], ecrãs de visualização de dados, e unidades integradas de comando e leitura de dados, associadas a autómatos. Essas unidades integradas são constituídas por botões e ecrã, ou, mais recentemente, um ecrã táctil resistivo e são capazes de apresentar dados em tempo real [24] (exemplo na figura 2.6). O futuro parece também promissor para a crescente utilização de tecnologias de realidade aumentada, nomeadamente como assistência à produção, manutenção e formação [40]

A programação de consolas HMI varia com os fabricantes, mas geralmente pode-se resumir a uma linguagem gráfica, *object-oriented*, em que o programador constrói o aspecto visual a

apresentar no ecrã, associando objectos apresentados com variáveis que interagem com o automático, para comando de diversas áreas de sistemas em modo manual, selecção de modos de funcionamento, apresentação de mensagens e alarmes do sistema, e armazenamento de informação.

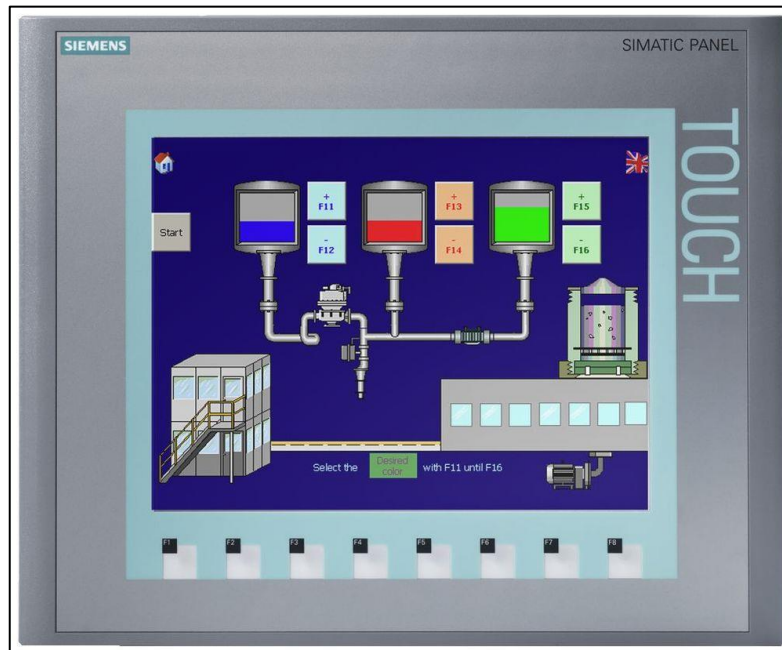


Figura 2.6: Exemplo de interface homem-máquina.

2.3.4. Sistema de Supervisão e Gestão de Produção

Os sistemas SCADA de supervisão e gestão de produção podem ser divididos em duas componentes essenciais: o servidor e os clientes. O servidor é uma unidade central de análise e tratamento de dados e toma a forma de um computador com o *software* de gestão necessário nele contido. Os clientes tratam da interacção entre o servidor e utilizador, sendo interfaces homem-máquina. A comunicação entre servidor e clientes é realizada através de uma comum ligação Ethernet TCP/IP ou outros métodos de comunicação, com excepção dos casos em que o computador onde o servidor esteja alojado também possua um HMI.

O servidor gere todos os dados recebidos dos sistemas a supervisionar, processando-os e armazenando-os numa base de dados local. Lendo dados de bases de dados locais – pode relacionar a informação das duas fontes (ordens de produção e a produção em curso), desse modo contabilizando e armazenando históricos de funcionamento do sistema a supervisionar. É também possível contabilizar e gerir alarmes do sistema.

Como interface entre o sistema de supervisão e gestão existem dispositivos denominados RTU – *Remote Terminal Unit* – que podem ser PLC's ou simples microcontroladores, que realizam o levantamento dos dispositivos de saída e entrada a que estão conectados, enviando os dados para o servidor e vice-versa (por exemplo, quando o sistema SCADA deseja enviar para o sistema de controlo ordens de produção).

Estes sistemas têm obrigatoriamente que ser robustos. Como tal, várias medidas são implementadas, nomeadamente redundâncias – redundâncias do local de armazenamento dos dados, por exemplo. Outra característica é a expansibilidade. Um sistema SCADA tem que estar preparado para, caso necessário, ser expandido, com a adição de clientes, RTU's ou servidores remotos [8].

Existe uma crescente preocupação com a segurança nestes sistemas, pois estão maioritariamente interligados com sistemas críticos na indústria ou mesmo nacionais (gestão de centrais nucleares, redes de electricidade e água potável, entre outros). Medidas para combater a falta de segurança incluem limitações de acesso ao sistema local ou remotamente, implementação de *firewalls* e desenvolvimento de mais e melhor segurança do sistema operativo do computador onde o servidor está alojado [41].

2.3.5. Sistemas de Visão Industriais

O crescimento dos sistemas de visão em ambiente industrial deve-se, em parte, à queda da relação custo por poder de processamento. Isto permitiu a criação de sistemas capazes de realizar uma aquisição com melhor qualidade, mais rapidamente e com um custo inferior. Estes sistemas são utilizados maioritariamente para efeitos de controlo de qualidade, relativamente a variações dimensionais, níveis de acabamento de superfícies e inspecções estruturais e operacionais (existência ou falta de elementos no produto) [42].

Estes dispositivos são compostos por vários componentes: um número variável de câmaras ou outros dispositivos capazes de captar imagens, um sistema de iluminação e uma unidade de processamento de imagem. As câmaras podem ser monocromáticas, policromáticas ou de infravermelhos, entre outras. A unidade de processamento pode ser remota – um computador, por exemplo – ou incluída na câmara, nas chamadas “*smart cameras*” [43].

A luz é um factor muito importante num sistema de visão. Dependendo da intensidade, ângulo de incidência ou espectro, uma análise pode ser bem ou mal realizada. Pouca luz dificulta o processamento da câmara, devido à sensibilidade do seu sensor, resultando em imagens com muito ruído. Luz directamente incidente em superfícies polidas acaba por ser quase ou totalmente reflectida para a câmara, podendo alterar a percepção da forma do componente a analisar, o que também dificulta a inspecção deste. É, portanto, extremamente importante a inclusão de

um sistema de iluminação que controle correctamente a luminosidade. Estes sistemas de iluminação podem incluir caixas negras, fontes de luz de vários tipos, painéis difusores, lasers, entre outros. Certos parâmetros podem ser também alterados nas câmaras ou na unidade de processamento, como o tempo de exposição do sensor destas primeiras, ou a variação do contraste e brilho no último [44].

2.3.6. Manipuladores Robóticos Industriais

Manipuladores robóticos são muito usados na indústria para realizar tarefas repetitivas, que necessitam de bastante rapidez e precisão, ou que limitam o acesso de pessoas a locais e processos perigosos ou que colocam em risco a saúde: soldadura, indústria química e metalúrgicas, por exemplo.

Tradicionalmente, a programação destes é efectuada “*online*”, ou seja, junto do robô, com uma consola de comando. Utilizando este método, o programador move manualmente o robô para pontos de interesse, armazenando a localização no espaço destes e definindo o movimento que o robô tem que efectuar entre dois pontos. Esta abordagem é relativamente directa para tarefas ou manipulação de objectos simples, mas à medida que o objecto a manipular se torna mais complexo, ou o espaço operacional é restringido por outros objectos ou estruturas circundantes, o programa a desenvolver torna-se muito mais complexo, necessitando de uma maior habilidade por parte do operador. Também existe a possibilidade de exposição do operador a ambiente hostis durante a programação *online*. Várias ferramentas e sensores existem para ajudar ao desenvolvimento do programa do manipulador através do método tradicional.

Outro método, em franca expansão, é a programação “*offline*”. Neste caso, recorre-se a *softwares* de simulação de células de fabrico e modelação tridimensional para criar virtualmente o espaço de trabalho, robô, objecto a manipular e ambiente circundante incluídos. Com a modelação concluída, pode-se utilizar a célula de fabrico virtual para criar e testar o programa do robô. Isto permite uma maior flexibilidade e controlo dos movimentos do robô, sendo mais acessível a programação de percursos bastante complexos. Também possibilita o desenvolvimento do programa do robô paralelamente ao desenvolvimento e construção mecânica da célula de fabrico. No entanto, o investimento inicial em licenciamento dos programas de desenvolvimento e em horas investidas a modelar a célula de fabrico pode não compensar para pequenas e médias empresas. Tecnologias de realidade aumentada começam a surgir para responder às limitações acima descritas da programação “*offline*” [45] [46].

Uma última técnica de programação existente designa-se por “programação através de demonstração”. Este método requer robôs especificamente criados para permitirem uma interacção directa com o programador ou operador (robôs colaborativos), já que este necessita de guiar fisicamente o manipulador robótico ao longo do percurso necessário, estando o robô a

armazenar continuamente a informação necessária para poder mais tarde repetir esses movimentos. É extremamente flexível, mas carece de alguma precisão. Pode ser utilizado em conjunto com o método tradicional para colmatar as limitações deste [47].

Os robôs são instrumentos que, por serem móveis, oferecem um perigo para a segurança de um indivíduo. Regra geral, são mantidos fora do alcance humano. Isto acarreta custos a nível de equipamento e espaço ocupado no chão da fábrica. Para combater estes custos, e oferecer a possibilidade de interacção entre pessoas e manipuladores robóticos, tem havido um desenvolvimento crescente de robôs colaborativos, mencionados no parágrafo anterior. Estes robôs estão dotados de dispositivos de segurança físicos e programados que lhes permitem realizar tarefas junto de um humano, seja em separado ou em colaboração (daí o nome), nomeadamente a detecção de esforços. Quando aplicados correctamente, têm a possibilidade de ser muito mais flexíveis nas tarefas desenvolvidas quando comparados com robôs tradicionais [48].

2.3.7. Sistemas de Identificação por Radiofrequências

Tecnologias RFID têm tido um crescimento muito grande na indústria e outros domínios. A implementação desta tecnologia tem uma série de benefícios, se implementados correctamente: maior eficiência operacional, redução de custos, melhor gestão de inventários e outros. É de especial interesse na automatização de manufactura de produtos, permitindo um acompanhamento personalizado de cada unidade ou conjunto de unidades, o que leva a maior flexibilidade no caso de um sistema que tenha que processar vários produtos com diferenças entre si ou distinguir entre um produto bom e outro defeituoso. A resistência à implementação do RFID é devida principalmente ao custo inicial ou crescente (relativamente à necessidade de compra contínua de *tags*) e dificuldades técnicas, seja devido à dificuldade de realizar *retrofitting* de sistemas existentes ou à sua implementação em sistemas novos [49].

As *tags* de RFID podem-se dividir em passivas e activas. A diferença principal entre estas é que as passivas utilizam a energia electromagnética transmitida pelo leitor RFID, enquanto que as activas incorporam uma bateria interna. A existência desta bateria permite a transmissão contínua da sua informação. Benefícios das *tags* passivas incluem um menor tamanho, custo muito baixo e tempo de vida longo. Já as *tags* activas têm um alcance de leitura muito grande quando comparado com as passivas e podem incluir sensores extra associados [50].

A comunicação entre *tags* e módulos de leitura/escrita é realizada, como o nome da tecnologia indica, por radiofrequências. Estas podem ser de baixa (LF, 125 kHz a 134 kHz), alta (HF, 13,56 MHz), frequência ultra alta (UHF, 433 MHz e 860 MHz a 960 MHz) ou frequência super alta (SHF, 2,45 GHz). Geralmente, frequências baixas e altas são mais apropriadas para aplicações de logística e traceabilidade de objectos ou produtos, com as de baixa frequência capazes de ser utilizadas em conjunto com materiais variados sem comprometer a integridade

do sinal, como produtos com alto teor de água ou têxteis, e as de alta frequência mais apropriadas para objectos metálicos. As frequências muito altas conseguem alcances maiores, mas o seu sinal é mais facilmente atenuado pelo ambiente e tem dificuldades em penetrar metal ou água [51].

2.4. Padrões de Programação e Análise de Programas

Recentemente, técnicas de modelação orientadas por objectos têm ajudado ao desenvolvimento de programas para sistemas de controlo industriais, permitindo o ataque de trechos complexos destes através dos conceitos de níveis de abstracção, encapsulamento e herança. Um *standard* destas técnicas é a UML – *Unified Modeling Language* [52]. Esta linguagem representa programas através de notação gráfica recorrendo a diagramas e consegue ser mais flexível que simples diagramas de sequências de operações, como o GRAFCET [53] [54].

Além de ajudarem ao desenvolvimento, ferramentas como a UML auxiliam também na análise de programas já desenvolvidos, permitindo a identificação de padrões e anti-padrões. Anti-padrões são pedaços de código reutilizados que, apesar de funcionarem correctamente, são impeditivos à boa organização ou eficiência de execução do programa, impedindo também uma fácil adaptação do código a novos problemas. Já padrões na programação são bons elementos de programas passíveis de serem reutilizados, devido à sua simplicidade e eficiência, podendo ser considerados como universais na possibilidade da sua aplicação em vários problemas [55]. Relativamente ao desenvolvimento de programas de sistemas de controlo, nomeadamente de autómatos, a procura de padrões e eliminação de anti-padrões é essencial, pois estes programas têm requisitos muito apertados quanto à eficiência e velocidade de execução. Devem-se procurar padrões em acções repetitivas do programa, como leitura e escrita de variáveis, accionamento de comandos a enviar para os periféricos, entre outros [56].

3. FERRAMENTAS E DISPOSITIVOS

3.1. Introdução

Neste capítulo apresentam-se as ferramentas e equipamentos utilizados neste estágio. Estes podem ser divididos em *software* e *hardware*. Quanto aos primeiros, temos os programas de desenvolvimento: de desenho de esquemas eléctricos, de programação dos vários dispositivos. Já quanto ao segundo, temos os dispositivos utilizados, desde os autómatos programáveis, a consolas de interface homem-máquina, passando pelos manipuladores robóticos, até ao sistema de visão. Apresentar-se-á também os métodos de comunicação utilizados.

3.2. Dispositivos e Equipamentos

Os sistemas de controlo utilizados no âmbito deste projecto baseiam-se em dispositivos Siemens. As unidades centrais de processamento das zonas de postos de trabalho são Siemens SIMATIC S7-300 CPU 315-2 PN/DP [57] (figura 3.1). Além destes, utiliza-se também módulos de cabeceira para interagir com as interfaces de comunicação, entradas e saídas e CPU, sendo o modelo SIMATIC ET200S DP IM 151-3 PN ST [58]. Os módulos de entrada são SIMATIC ET200S DP 8DI DC24V [59] e os de saída SIMATIC ET200S DP 8DO DC24V/0,5 [60], utilizando-se módulos de potência SIMATIC ET200S DP PM-E DC24V [61] para alimentação dos módulos de I/O. Para as consolas HMI destas zonas utilizam-se Siemens KTP700 Basic [62]. Relativamente ao sistema de transporte entre o posto de controlo de pinhão e o OP90, o PLC a usar será um SIMATIC ET200S DP IM 151-8 PN/DP [63] e um HMI TP700 Comfort [64]. Outro autómato igual será utilizado como RTU do sistema SCADA.

Relativamente ao sistema SCADA, este é baseado num computador industrial, com um módulo de expansão da Siemens para comunicação directa via PROFINET (ver capítulo 3.4 – Interfaces de Comunicação) com autómatos e sistema de armazenamento em RAID 1 para redundância. As suas interfaces homem-máquina são um ecrã, rato e teclado.

Equipamentos adicionais do sistema incluem *drives* de motores Rexroth IndraDrive [65] [66], prensas eléctricas DSM SMP 500/50 200 V2 DR 00 00 de 50kN [67], aparafusadoras baseadas em sistema da SOPAP [68], leitores de códigos de barra Keyence SR-1000 [69] e sistemas

de monitorização Kistler maXYmos BL [70]. Também se utiliza um indexador rotativo Weiss TC220 [71] no OP25.

O sistema de visão é composto por uma *smart camera* Teledyne Dalsa BOA BVS-1280M-INS [72] (figura 3.2). Esta câmara tem ligações I/O digitais e Ethernet, suportando uma variedade de interfaces de comunicação. Os manipuladores robóticos são da ABB e Universal Robots, utilizando-se o modelo UR10 [73] (figura 3.3) para o OP25 (um modelo colaborativo), o modelo IRB 2600 [74] (figura 3.4) para a OP60 e OP90.1, o modelo IRB 1600 [75] para o OP70.1 e IRB 4600 [76] para os OP90.2 e OP90.3. O robô da Universal Robots possui seis graus de liberdade para seis juntas, enquanto que os robôs ABB têm todos seis juntas e seis mais três graus de liberdade: um para cada junta com três extra para estruturas móveis auxiliares, externas ao braço. As diferenças entre os três modelos ABB são não apenas as dimensões destes, mas também a carga máxima suportada no punho (incluindo manipulador).

Os leitores/escritores RFID são constituídos por módulos “mestre” IFM DTE101 [77] que comunicam com os PLC, com cabeças de leitura/gravação (antenas) ligadas a estes, o que permite um posicionamento destes “escravos” com distâncias consideráveis entre estes. O sinal rádio utilizado tem frequência de 13,56 MHz (HF). As *tags* utilizadas são passivas.



Figura 3.1: Siemens SIMATIC S7-300 CPU 315-2 PN/DP.



Figura 3.2: Teledyne Dalsa BOA BVS-1280M-INS.



Figura 3.3: Robô ABB IRB 2600.

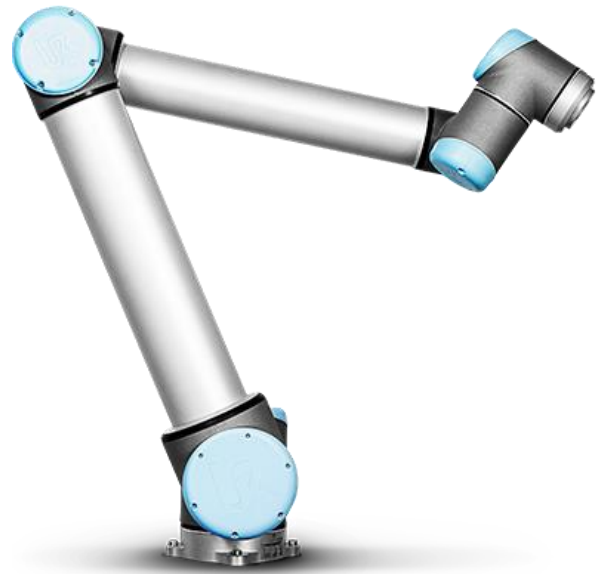


Figura 3.4: Robô Universal Robots UR10.

3.3. *Software* Utilizado

Tantos componentes programáveis diferentes obrigam à utilização de uma também grande variedade de programas. Nesta secção vão ser apresentadas as ferramentas de *software* a utilizar para desenvolvimento.

3.3.1. Programas de Desenvolvimento Siemens

A Siemens disponibiliza um software intitulado TIA Portal [78] [79] (figura 3.5) que integra a programação de todos os dispositivos de automação comercializados por esta empresa: autómatos, a nível do seu programa e configuração do hardware e comunicação entre estes e os seus periféricos [80], programação de consolas HMI, criação e desenvolvimento de sistemas de gestão SCADA, programação de drives de motores, planeamento, parametrização e diagnóstico de motores e fontes de alimentação, entre outras possibilidades. É uma ferramenta muito poderosa e flexível devido ao facto de oferecer um ambiente de desenvolvimento comum aos produtos Siemens.

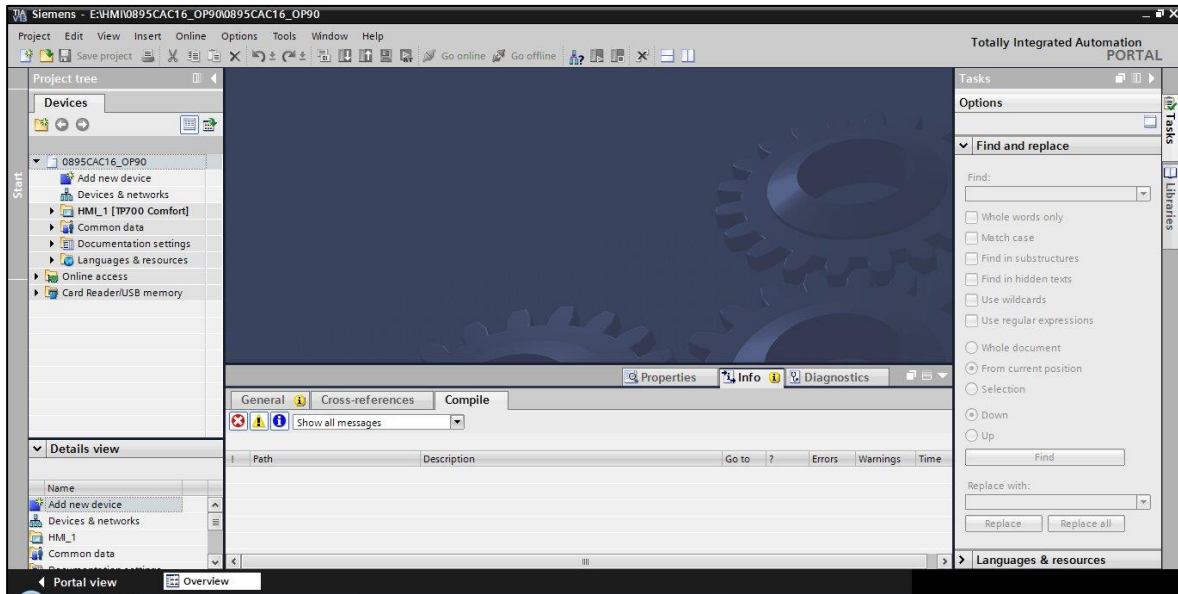


Figura 3.5: Ambiente gráfico do TIA Portal V12.

Para um desenvolvimento não integrado, existe o STEP 7 [81] (figura 3.6), programa de desenvolvimento para autômatos e o WinCC [82], para programação de sistemas de gestão e supervisão industriais. O primeiro oferece um ambiente gráfico simples, com uma organização por árvore. A programação de autômatos pode ser realizada recorrendo a linguagem *Ladder*, FBD ou STL, um dialecto de *Instruction List* próprio da Siemens [83]. É possível a utilização de todas as linguagens em simultâneo num mesmo programa, se necessário. É menos intuitivo quando comparado com o TIA Portal, mas muito mais leve relativamente a recursos computacionais. Já o WinCC (figura 3.7) oferece a possibilidade de programação tanto do servidor como do cliente num sistema SCADA, incluindo vários métodos de comunicação com periféricos e com servidores remotos. O WinCC suporta *scripting* utilizando como linguagens dialectos de VB Script e ANSI C. As linguagens e métodos de programação nos *softwares* não integrados e no integrado são exactamente as mesmas. Todos os programas apresentados dispõem de métodos de simulação dos programas neles desenvolvidos.

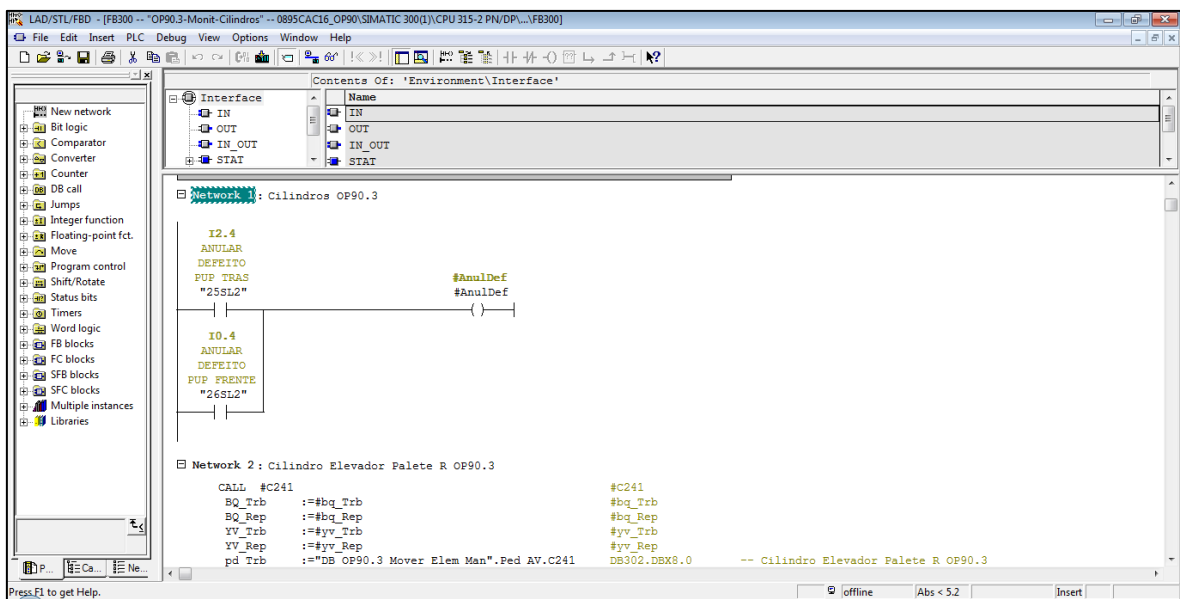


Figura 3.6: Ambiente gráfico do STEP 7 V5.5, janela de projecto.

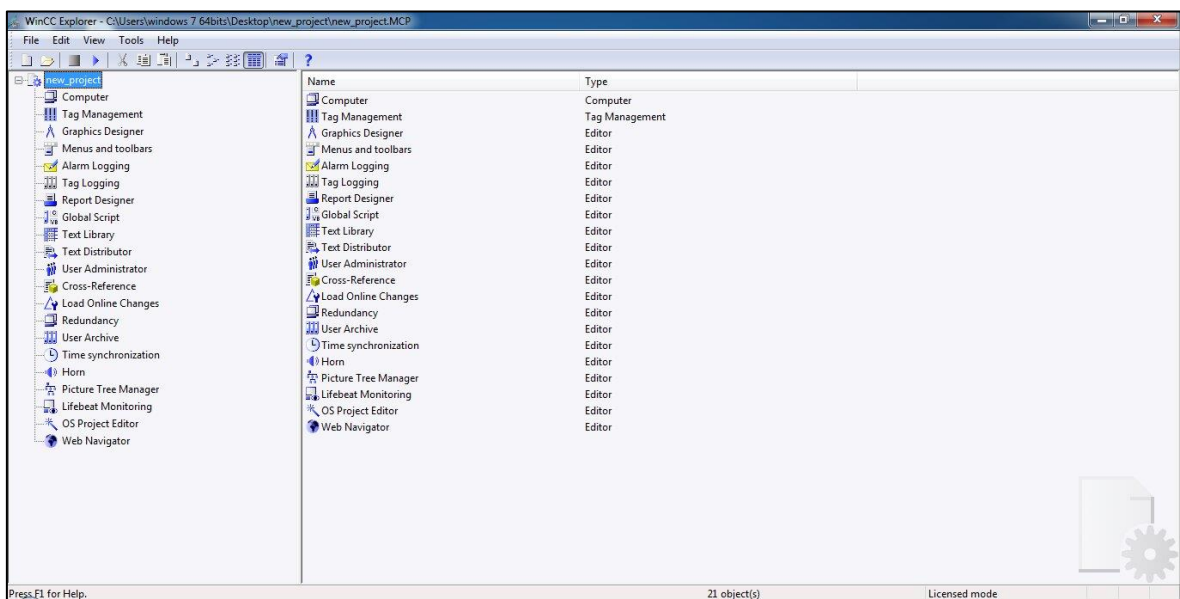


Figura 3.7: Ambiente gráfico do WinCC V7.3.

Os robôs ABB podem ser programados utilizando a linguagem de programação RAPID [86], sendo que esta programação pode ser *online* ou *offline*. Para programar *offline*, pode-se recorrer ao RobotStudio [87] (figura 3.10), uma ferramenta de simulação e modelação de células de fabrico. Já o braço robótico UR10 é programado *online* através de uma consola. Seja num caso ou no outro, o candidato não programou directamente os manipuladores utilizando o seu software, tendo-se limitado a programar o funcionamento destes do lado dos autómatos.

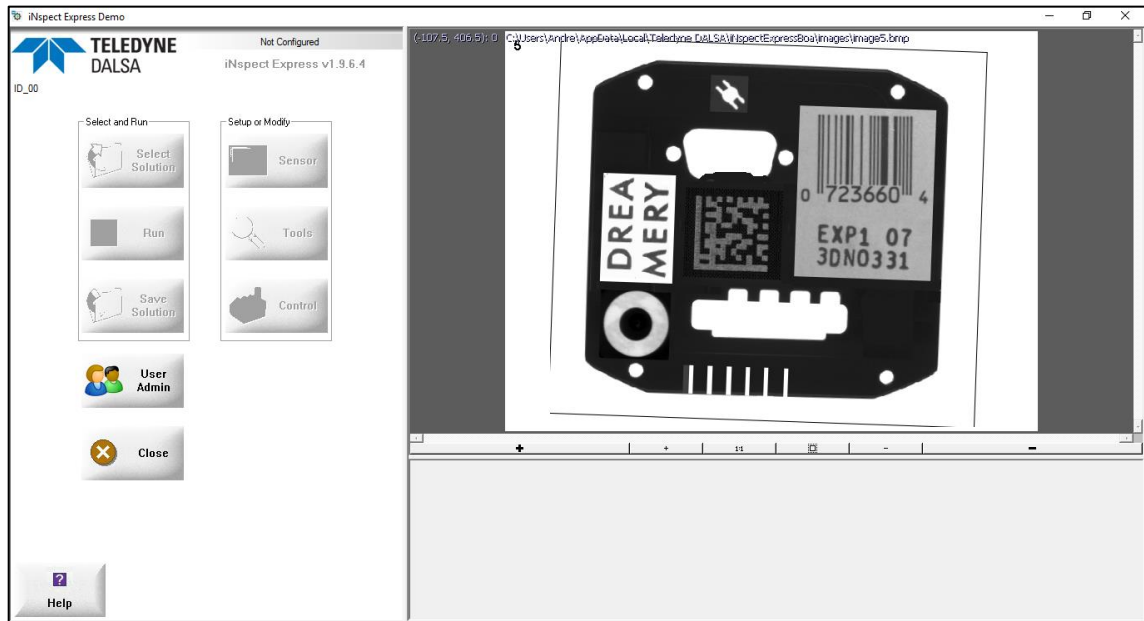


Figura 3.9: Ambiente de desenvolvimento do iNspec Express. Simulação a correr.

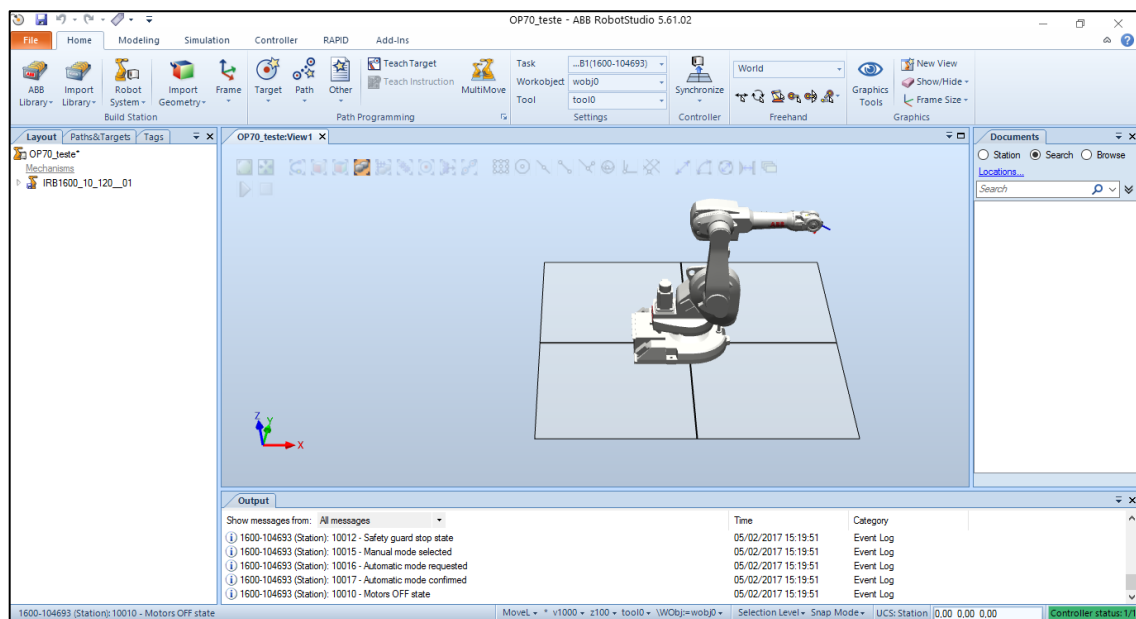


Figura 3.10: Ambiente gráfico do RobotStudio. Célula de Fabrico vazia com robô IRB 1600 modelado.

Para programação de prensas eléctricas DSM existe o *software* DSM MultiControl4. Este programa permite não só a configuração da prensa, seja das acções a executar por esta (controlo posicional e de esforço) ou a sua comunicação com o autómato, como também a sua monitorização em tempo real a partir do computador. Um exemplo de programa para a prensa e ambiente gráfico do MultiControl4 pode-se ver na figura 3.11.

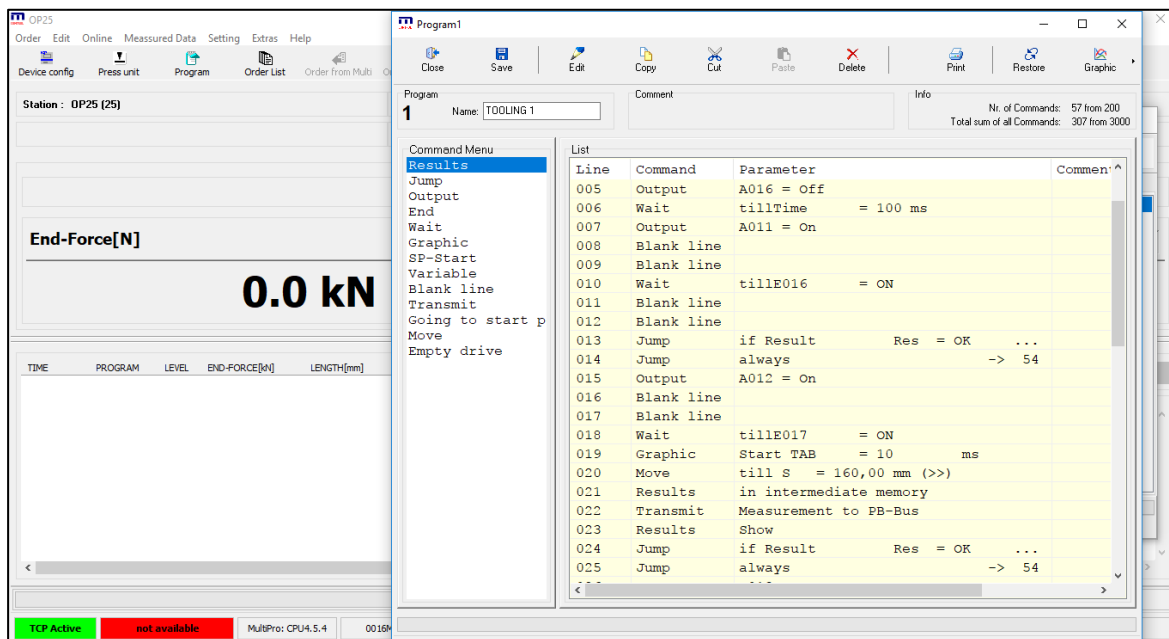


Figura 3.11: Ambiente gráfico do DSM MultiControl4.

3.4. Interfaces de Comunicação

As interfaces de comunicação utilizadas basearam-se em três tecnologias: PROFIBUS, PROFINET IO e TCP/IP.

A interface de comunicação PROFIBUS é um protocolo que define uma rede de múltiplos níveis. Consiste numa linha de comunicação central (barramento), com a qual os dispositivos interagem, comunicando deste modo entre si, com um dos equipamentos agindo como *master* e os restantes como *slaves*. Um segundo método de comunicação com PROFIBUS baseia-se num sistema de “*token passing*”, em que um sinal “*token*” pode ser transmitido entre diferentes *masters*. Neste caso, apenas o dispositivo com o “*token*” pode comunicar com os restantes equipamentos, agindo como *master* [88].

PROFINET IO é um protocolo de comunicação construído sobre tecnologias existentes de Ethernet. O princípio utilizado é o mesmo que no PROFIBUS, mas neste caso a velocidade de transmissão é muito mais elevada. Esta tecnologia emprega três canais de comunicação

entre dispositivos: um canal TCP/IP, para configuração da comunicação, outro denominado RT – *Real Time* – para transferência normal de dados e alarmes dos sistemas e um terceiro, IRT, ou RT síncrono, para aplicações de controlo de movimentos. Os dois últimos canais mencionados são mais rápidos a transmitir informação comparados com TCP/IP, pois a prioridade da informação transmitida nestes canais é mais elevada, além de a transmissão ser em tempo real. O IRT, especificamente, possui mecanismos próprios de reservas que garantem ciclos com um *jitter* inferior a um microssegundo.

A configuração destas duas tecnologias é realizada através da configuração de drivers, instalado através de ficheiros chamados GSD – *Generic Station Description*. Habitualmente, estes ficheiros são importados para o programa de desenvolvimento para autómatos, e aí configurados [89].

Relativamente à comunicação TCP/IP, esta irá ser utilizada principalmente para aplicações que não exijam que a comunicação seja em tempo real, como por exemplo para comunicação entre o sistema SCADA e o servidor da unidade fabril [90].

4. SISTEMAS DE VISÃO

4.1. Introdução

No posto de trabalho OP70 da linha de montagem, após montagem do rotor com paletas e anilha superior, é necessário verificar que estes elementos foram correctamente montados na bomba de óleo. Para tal foi definido pela empresa a utilização de um sistema de visão por computador a implementar no posto OP70.5. Também foi desenvolvido um segundo sistema de visão, que será abordado adiante. A menos que seja explicitamente dito, toda a informação neste capítulo é referente ao sistema de visão do posto OP70.5 da linha de montagem em desenvolvimento.

Mecanicamente, este posto é composto por uma caixa negra, de fundo aberto – projectada de modo a impedir ao máximo a influência de luz exterior – contendo o sistema de visão, com o sistema transportador de bombas de óleo por baixo da tal caixa. A paleta com a bomba, ao chegar ao posto de trabalho, é travada directamente por baixo da caixa, impedida de avançar mais, e ligeiramente elevada por um cilindro, fixando-a na posição de análise. Este cilindro garante que todas as bombas são colocadas no mesmo local relativamente ao sistema de visão, evitando ao máximo diferenças de posição das bombas de óleo nas imagens captadas pelo sistema. Por fim, após a paleta ser elevada, as entradas e saídas de paletes do posto são bloqueadas por painéis que descem, reduzindo desse modo ainda mais a captação de luz exterior pelo sistema de visão, seguindo-se a inspecção da bomba. Após realizada a inspecção, os painéis laterais sobem, o cilindro de bloqueio desce e a paleta segue para o próximo posto, caso a análise tenha identificado todos os elementos bem colocados, ou para o local de saída de bombas defeituosas, caso contrário. Um esquema do posto pode ser observado nas figuras 4.1 e 4.2.

O sistema de visão propriamente dito é composto por uma *smart camera* Teledyne Dalsa BOA, possibilitando a aquisição e processamento de imagens monocromáticas de resolução 1280x960, possibilidades de comunicação através de entradas e saídas digitais ou cabo Ethernet [91], e um sistema de iluminação, um anel de LED's de luz vermelha. A selecção destes elementos foi determinada pela empresa.

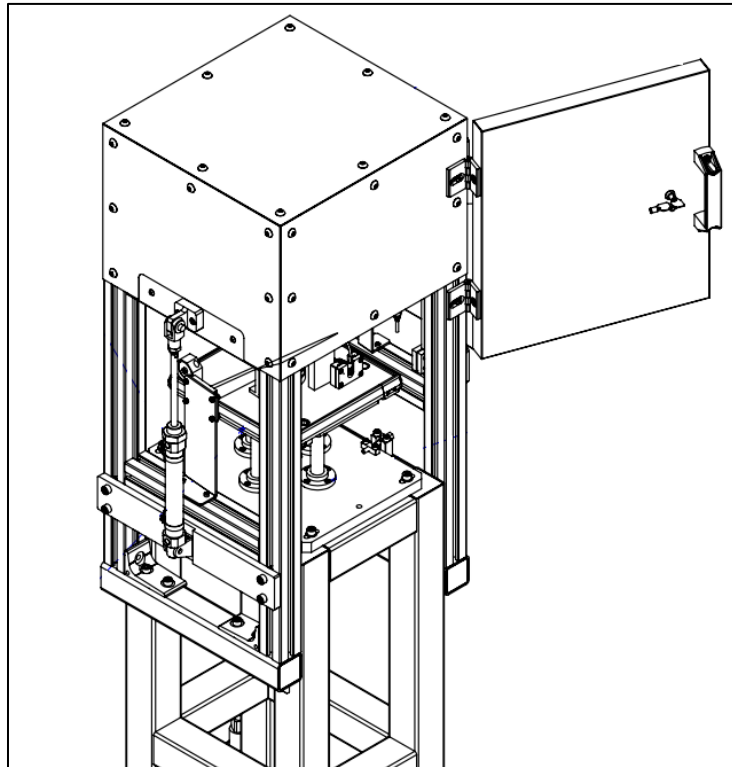


Figura 4.1: Esquema do posto de trabalho OP70.5, vista isométrica.

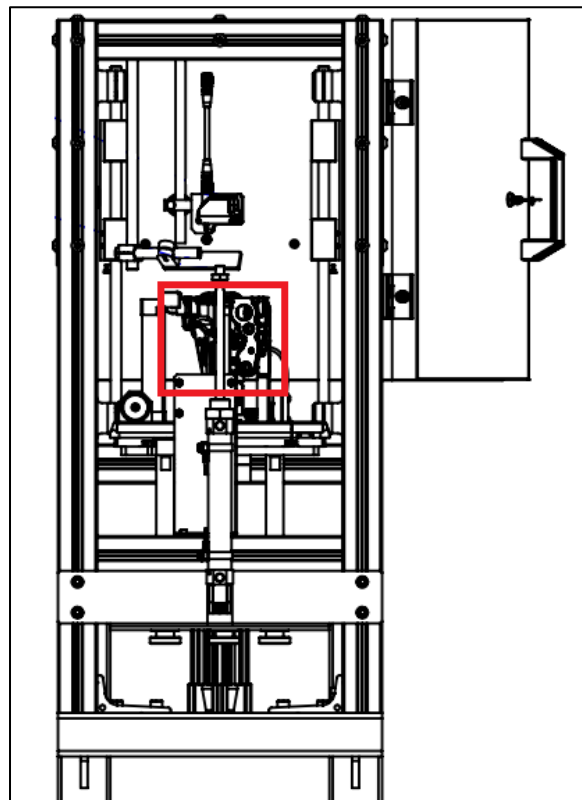


Figura 4.2: Esquema do OP70.5, vista lateral, com bomba posicionada (dentro do quadrado).

4.2. Requisitos do Sistema

Como dito anteriormente, o objectivo principal é a análise da existência e bom posicionamento de vários elementos da bomba de óleo: as sete palhetas e uma anilha no rotor. É necessário que o sistema seja capaz de identificar correctamente tanto palhetas polidas como palhetas “escuras”, presentes no modelo M9T. Adicionalmente, analisa-se também a existência ou ausência do segmento metálico entre o estator e o corpo da bomba. Na figura 4.3 vê-se uma anilha, dois segmentos, visto de lado, e um segmento, também ele visto de lado.

Outro requisito é a interacção com os restantes sistemas da linha. O sistema de visão tem que ser capaz de interagir com o autómato que controla o OP70, recebendo ordens de análise e enviando os resultados (um por cada análise), assim como interagir com o servidor, enviando para o computador que o aloja registos das análises, tanto imagens como históricos detalhados.

É crítico que este sistema tenha a precisão e robustez suficiente para realizar ininterruptamente análises ao longo de milhares de horas, e que essas análises efectuadas sejam correctas, ou seja, que o resultado coincida sempre com a realidade.



Figura 4.3: Anilha, palhetas e segmento.

4.3. Esquema Eléctrico do Sistema de Visão

O sistema de visão, a nível eléctrico, consiste apenas na ligação da câmara ao autómato e à alimentação, assim como a ligação do sistema de iluminação. A ligação da câmara contou com a verificação da cablagem a utilizar, alimentações disponíveis no quadro eléctrico do OP70 que pudessem alimentar a câmara (24V DC) e local apropriado a conectar a câmara ao PLC para comunicação. Já o sistema de iluminação, alimentado também a 24V, é apenas ligado quando o cilindro elevatório da paleta é accionado, utilizando-se um relé para activar a iluminação. Isto garante um baixo consumo energético, estando o sistema desligado quando não está em uso.

Na figura 4.4 pode-se observar a página do esquema eléctrico do posto OP70 contendo as ligações da câmara. Esta está do lado esquerdo, com os seus conectores representados ao centro e caixas informativas à direita apresentando as cores dos conectores individuais dos cabos.

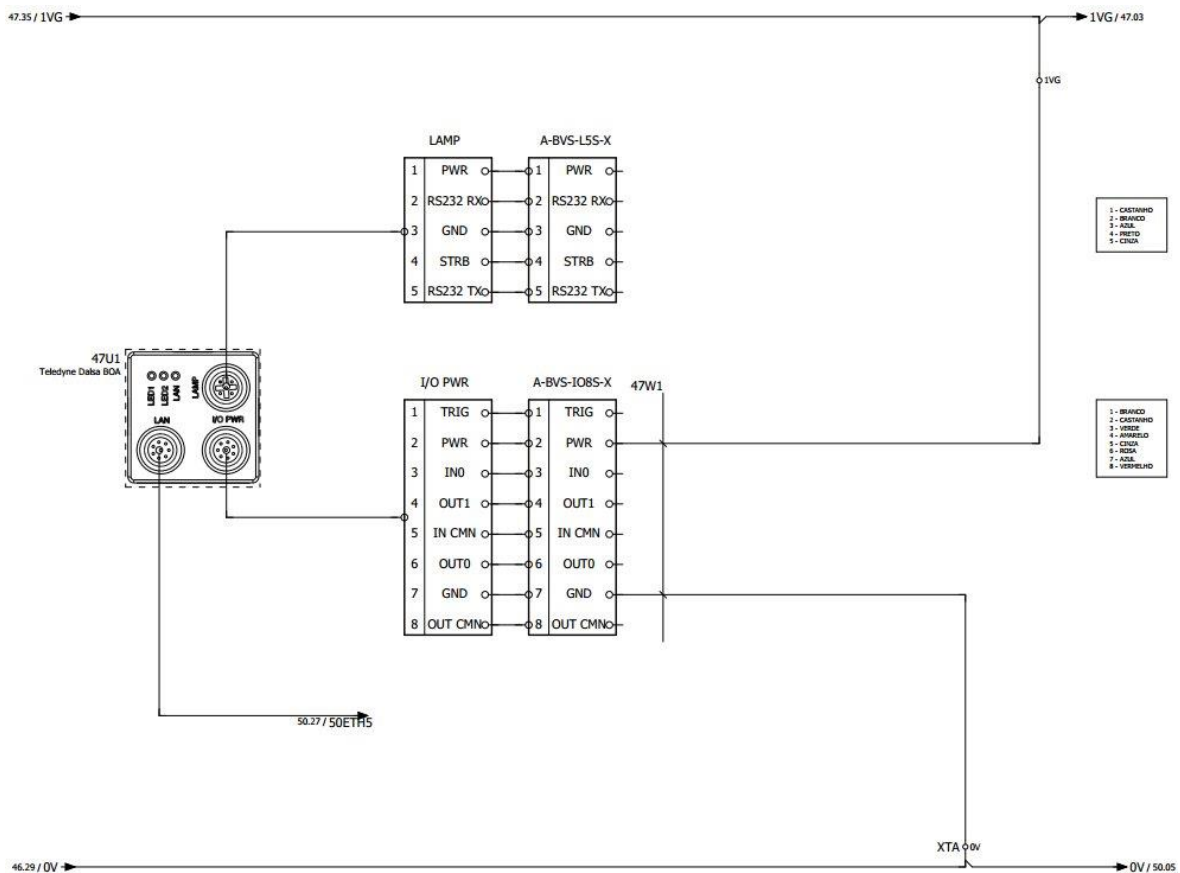


Figura 4.4: Esquema das conexões da câmara Dalsa BOA.

4.4. Ajuste Mecânico do Sistema

O primeiro passo no desenvolvimento e implementação de um sistema de visão industrial é o ajuste mecânico do sistema. Este envolve o posicionamento espacial da câmara e do sistema de iluminação, assim como o ajuste da óptica da primeira. Através de veios e suportes que podem ser colocados ao longo desses veios, posicionou-se a câmara de modo a ter o seu sensor de grosso modo normal ao topo da bomba de óleo. A distância entre a câmara e o topo da bomba foi ajustado de modo a que a zona de interesse (estator e rotor) ocupasse a maioria da imagem, já que rentabilizando o número de pixéis da imagem facilita a análise de zonas (objectos, gradientes) pequenos. O anel de iluminação foi posicionado também através de veios de modo a que a luz que chegasse à superfície da bomba fosse uniforme, limitando a existência de reflexos indesejáveis e criando um contraste elevado entre objectos e áreas circundantes. Objectos bem delimitados facilitam o trabalho de análise e processamento do sistema de visão, diminuindo a necessidade da utilização de algoritmos de processamento pesados computacionalmente. Na figura 4.5 pode-se observar a disposição da câmara (pequeno quadrado no centro) e anel de iluminação (em redor da câmara), fixados por suportes e veio (o veio de suporte da iluminação encontra-se directamente por baixo do veio de suporte da câmara, e por isso não é possível ser observado). A bomba encontrar-se-ia na paleta (que contém uma seta, número de paleta e modelo de bomba gravados), centrada, por baixo do sistema de iluminação. Pode-se observar também uma foto (figura 4.6) do processo de posicionamento da iluminação e a distância relativa entre esta, a câmara e a bomba de óleo na paleta.

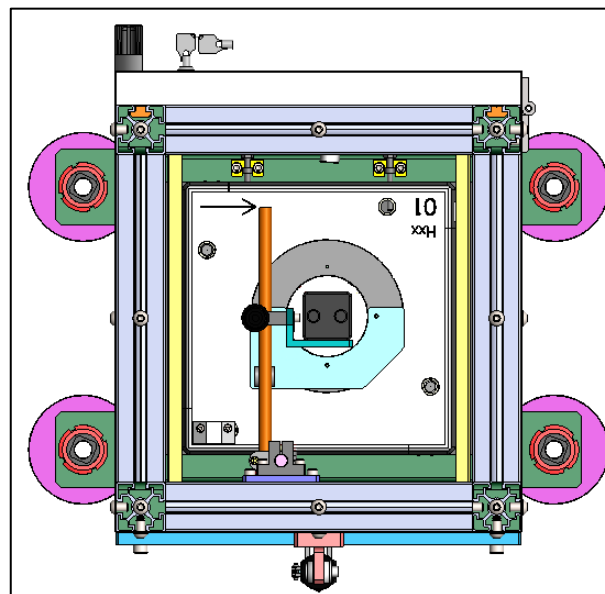


Figura 4.5: Esquema do posicionamento do sistema de visão no posto OP70.5, visto de cima.

Quanto ao ajuste óptico da câmara, foi possível variar cinco parâmetros, tanto a nível mecânico como através de software, de modo a conseguir-se a captura de imagens o mais claro e definido possível: abertura da lente, distância focal, tempo de exposição do sensor e processamento da imagem ao nível do brilho e contraste. Inicialmente, através da utilização do software iNspec Express, configurou-se o brilho e contraste para um nível intermédio (50%, de 0% a 100%). Também se configurou o tempo de exposição para um valor intermédio (26,00 ms, sendo a gama possível de 0,05 ms a 51,10 ms). A configuração destes parâmetros nos seus valores intermédios permite a maior gama possível de ajustes após serem configurados os parâmetros mecânicos. Seguiu-se a parametrização da óptica: procurou-se inicialmente focar a câmara, variando a distância focal (em mm), de modo a que a superfície superior da bomba (a zona de interesse) ficasse focada, em detrimento do resto da imagem. Com a imagem focada, seguiu-se o ajuste da abertura da lente (em mm), de modo a permitir a chegada de luz suficiente ao sensor da câmara, não comprometendo a definição da imagem. A relação entre estes dois parâmetros é chamada relação focal, ou “número F ”, e é descrita pela equação 4.1 [92]:

$$F = f/D \quad (4.1)$$

Sendo:

F a relação focal, habitualmente representada no formato “ f/F ”;

f a distância focal;

D a abertura de lente.

Estes ajustes envolveram também pequenas mudanças na posição do sistema de iluminação, para garantir a imagem com melhor qualidade possível. O parâmetro de contraste revelou-se pouco útil, não causando diferenças perceptíveis na imagem, e, portanto, ajustou-se apenas o tempo de exposição do sensor e o brilho da imagem. Pode-se observar na figura 4.7 um mosaico de imagens com estudos do tempo de exposição e brilho, com a abertura de lente e distância focal constantes, e na figura 4.8 um exemplo das imagens captadas após ajuste óptico da câmara. Os parâmetros finais do sistema de visão são:

- Abertura de lente: 6 mm;
- Distância focal: 180 mm;
- Relação focal: $f/30$;
- Tempo de exposição: 7,80 ms;
- Brilho: 46%;
- Contraste: 50%.

Para garantir que a iluminação e parametrização da câmara ficassem suficientemente bem calibradas, testaram-se situações impossíveis e extremas, como incidir directamente a luz de um foco no topo da bomba de óleo, verificando se tal alterava a imagem significativamente. Chegou-se à conclusão que o sistema estava devidamente ajustado.

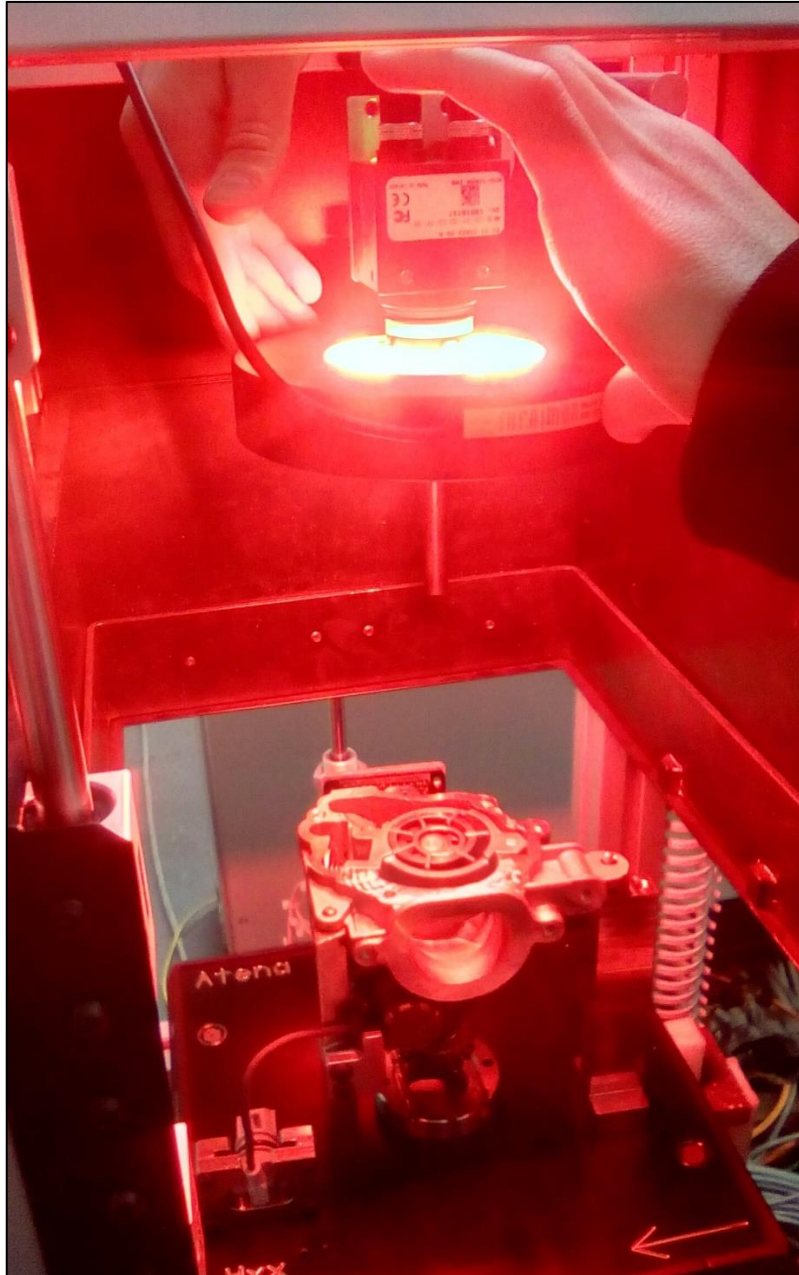


Figura 4.6: Posicionamento da iluminação no sistema de visão.

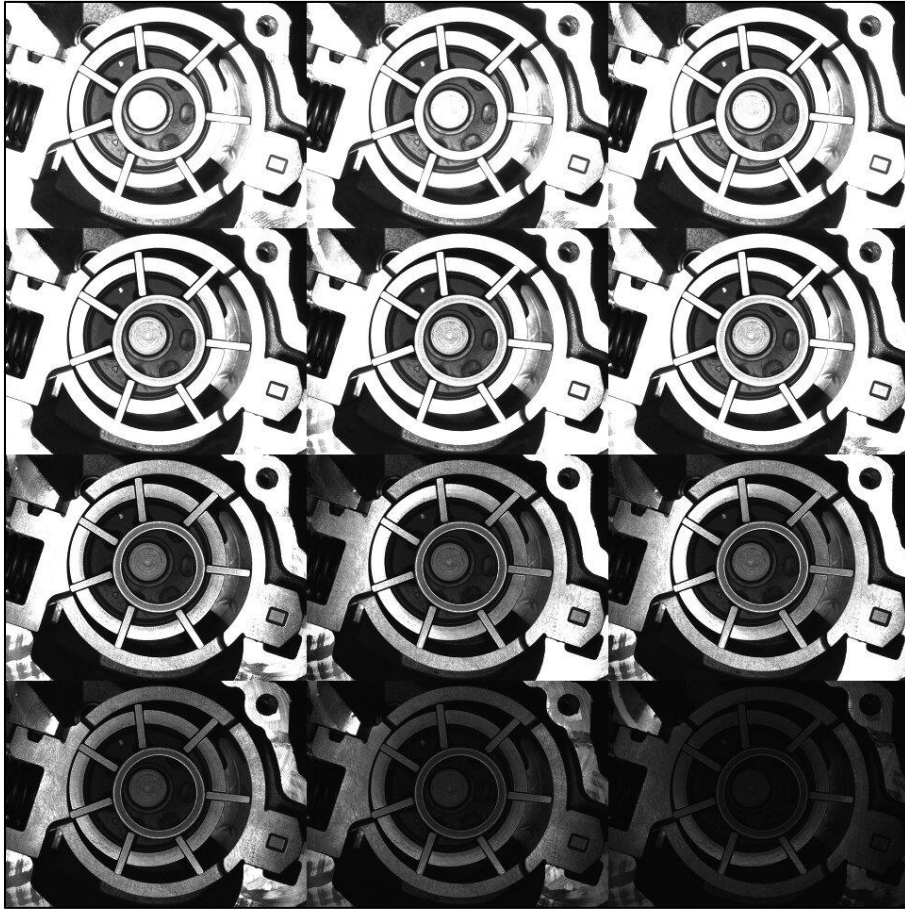


Figura 4.7: Estudo do tempo de exposição e brilho.

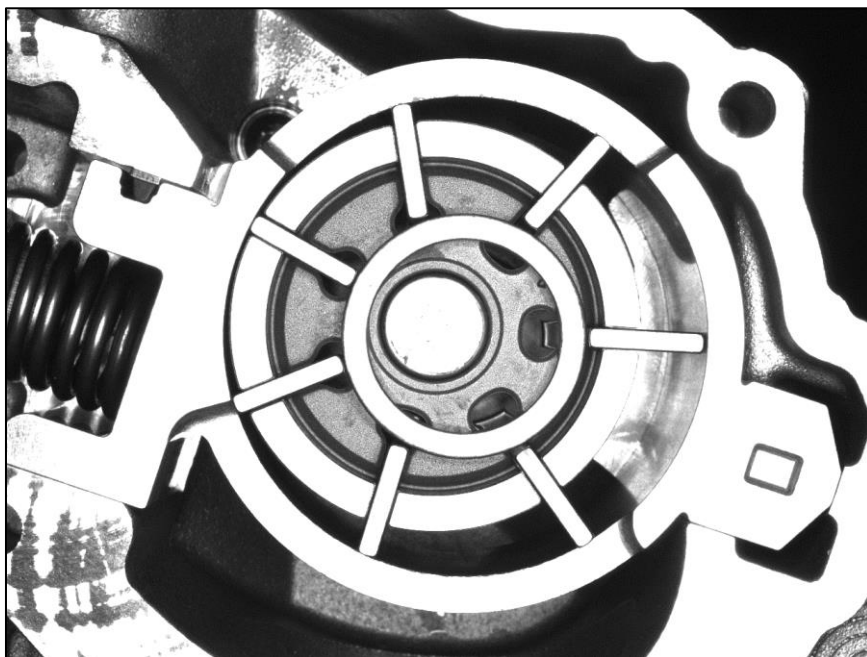


Figura 4.8: Exemplo de imagem captada após ajuste óptico e de posição do sistema.

4.5. Programa de Inspeção de Bombas de Óleo

Para cumprir os requisitos do sistema de visão, foi necessário desenvolver dois programas: um primeiro para análise das imagens captadas pela câmara e registo dos resultados, e um segundo para comunicação e interacção entre a câmara e o autómato do posto de trabalho. O programa de análise foi desenvolvido utilizando o *software* iNspect Express, tendo o do autómato sido criado com o Step7. Nesta secção expor-se-á o primeiro programa.

4.5.1. Imagem Padrão

Para se poder realizar uma análise a uma imagem é necessário, antes de tudo, criar uma relação de comparação. É impossível, para um sistema, conseguir analisar uma imagem e identificar regiões de interesse com precisão se este não tiver algo em que se basear. Como tal, foi captada uma imagem de uma bomba de óleo (apresentada na figura 4.8) para que esta servisse de padrão. Sendo o padrão, é importante que esta imagem contenha todas as palhetas presentes e a anilha bem colocada, assim como o segmento. Resumidamente, é necessário que a imagem padrão seja a de uma bomba não defeituosa, sendo que qualquer imagem que não corresponda ao padrão pertence a uma bomba de óleo defeituosa.

4.5.2. Localizador

A segunda preocupação é o facto de a posição da bomba relativamente ao sensor da câmara poder não ser constante, ou seja, é possível, mesmo com o cilindro de bloqueio da paleta elevado, que existam ligeiras diferenças posicionais, no espaço. Isto pode ocorrer devido a pequenas diferenças nas paletas, ou mesmo variações dimensionais da bomba. O *software* de desenvolvimento oferece a possibilidade de ter em conta estas variações, utilizando localizadores. Localizadores são obtidos através de pontos retirados da análise de uma região de interesse (ROI – *Region of Interest*) na imagem. A título de exemplo, imagine-se que, nas imagens a analisar, se tem um objecto facilmente identificável através da comparação de pixéis entre estas e a imagem padrão. É possível identificar previamente o objecto na imagem padrão e identificar o seu centróide, e consequentemente identificar objectos semelhante noutras imagens e retirar também os seus centroides. Comparando a posição relativa do centróide “padrão” em relação aos restantes, é possível ter em conta essa diferença posicional e deslocar outras ROI definidas na

imagem padrão destinadas a identificar outros objectos para identificar correctamente os mesmos objectos em novas imagens. Com um ponto apenas é apenas possível ter em conta diferenças posicionais, mas, se se tomar em consideração dois pontos, pode-se formar um vector, relacionar o ângulo desse vector com, por exemplo, o eixo horizontal da imagem, e obter informação acerca da diferença de rotação entre a imagem padrão e outras. É muito difícil (e impossível, com este equipamento) ter em conta variações na distância entre a bomba e o sensor da câmara. Tais variações têm que ser tomadas em conta durante a definição de ROI subsequentes à do localizador.

Para definir o localizador no presente problema procurou-se uma característica constante em todas as bombas de óleo e extremamente fácil de identificar com precisão, para evitar erros de posicionamento acumulados ao longo das várias análises (identificação de objecto, identificação dos pontos de localização, posicionamento de ROI subsequentes, identificação de objectos dentro dessas novas ROI). Torna-se óbvio que o melhor candidato é a zona de contacto entre o estator e a mola (no lado esquerdo da figura 4.8). Esta zona apresenta pelo menos dois segmentos de recta formados por gradientes de pequena largura, mas muito fortes (identificados na figura 4.9). Utilizou-se a ferramenta de “*match*” do *software* para comparar pixel a pixel esta zona entre a imagem padrão e novas imagens. Definiu-se para esta ferramenta uma zona considerável de procura e efectuou-se uma binarização nesta zona, de modo a exacerbar a passagem de preto a branco. Também se excluiu a zona central da mola da análise, para evitar que certos reflexos desta pusessem em causa a análise (“a”, na figura 4.10, que apresenta a área binarizada a procurar). Todas as regiões de interesse depois definidas foram então associadas a este localizador.

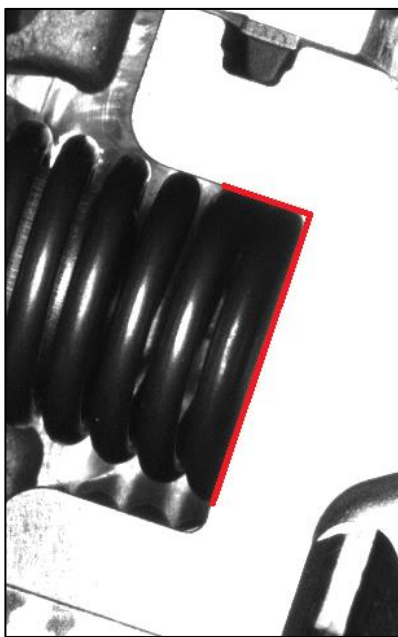


Figura 4.9: Gradientes de interesse para localizador.

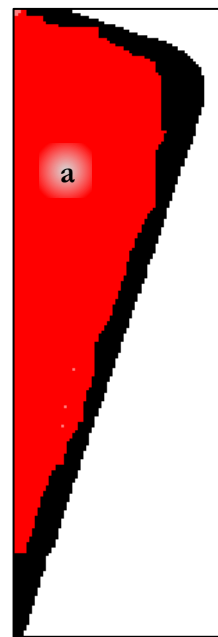


Figura 4.10: Área de interesse para localizador.

4.5.3. Identificação da Anilha

Utilizando a ferramenta de identificação de elementos circulares do *software* de desenvolvimento, é fácil identificar a anilha, colocada em cima do rotor. Para configurar esta ferramenta é necessário definir um círculo de um diâmetro pré-definido, assim como dois círculos de diâmetros inferior e superior ao primeiro, sendo que a ROI de procura do diâmetro do primeiro círculo é a área definida por estes dois círculos. A câmara processa então essa região de interesse, procurando, desde o círculo interior até ao círculo exterior, ao longo de um número de linhas distanciadas igualmente entre si, um gradiente (neste caso, de escuro para claro, ou seja, a passagem do fundo para o diâmetro interior da anilha). Encontrando todos os pontos que correspondam a gradientes fortes, forma com eles um círculo, calculando depois o seu diâmetro, em pixels. Caso o valor do diâmetro se encontre dentro de um limite máximo e mínimo especificados, considera-se que se encontrou a anilha. Esta abordagem é possível devido ao facto de, como se pode observar na figura 4.8, o gradiente mais forte de diâmetro próximo do do círculo interior da anilha ser exactamente esse. Para eliminar falsas identificações de pontos que não pertençam ao diâmetro interior da anilha é aplicada uma binarização com um limiar muito elevado.

Caso a anilha não se encontre presente e sejam identificados pontos suficientes para formar um círculo, por mais remota que seja a possibilidade, este dificilmente terá o diâmetro igual ao da anilha. No entanto, para garantir que não ocorrem más análises (identificações de anilha quando esta na verdade não existe), realiza-se uma análise de três passos, sendo que um deles é a identificação comentada anteriormente. A segunda é a identificação, utilizando exactamente a mesma técnica (mas procurando uma passagem de claro para escuro), do diâmetro exterior do estator. Esta identificação é útil, já que se pode observar que a anilha e o exterior do estator são, de grosso modo, concêntricos. Tendo sido identificados com sucesso ambos os elementos, basta testar a concentricidade destes – a terceira análise – para garantir que o sistema consegue diferenciar entre uma anilha bem colocada, uma anilha existente mas mal posicionada (por exemplo, uma anilha pousada sobre uma palheta) ou a falta de anilha mas má identificação, ou falso positivo, do seu diâmetro, e a falta de anilha.

Na figura 4.11 pode-se observar a área de procura do diâmetro da anilha, entre ambos os círculos azuis, as linhas de procura de gradientes a amarelo, o diâmetro encontrado a rosa (que corresponde praticamente ao diâmetro interno da anilha) e o seu valor, em pixels, com o valor “óptimo” entre parêntesis, e os centros dos círculos interior da anilha e do estator, como pontos amarelo e verde, respectivamente.

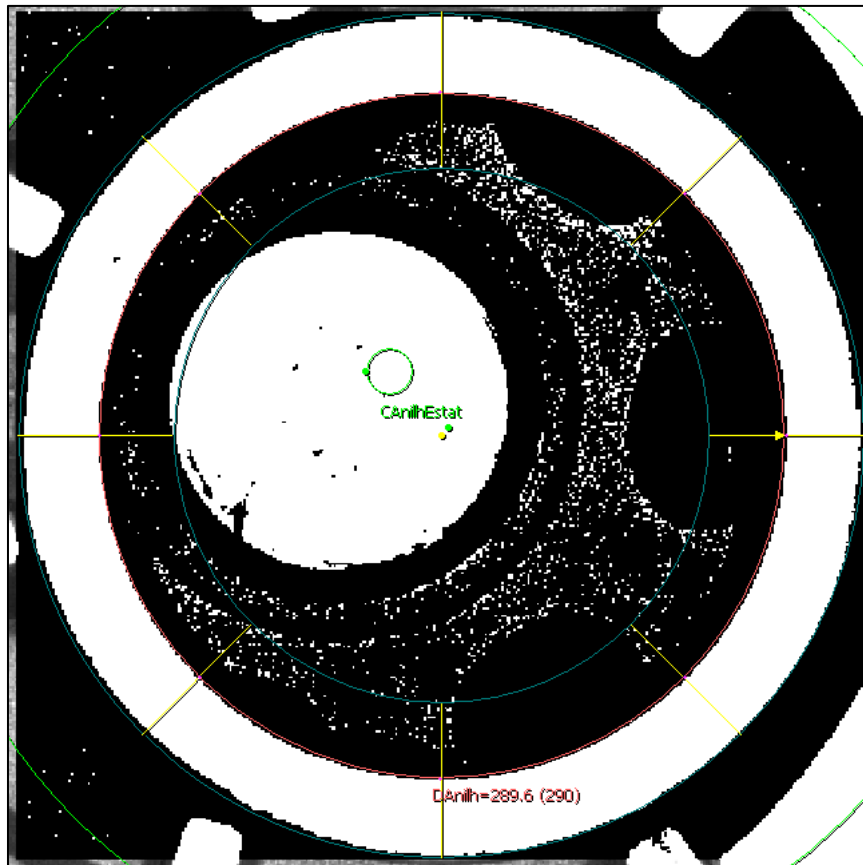


Figura 4.11: Análise da existência de anilha na bomba de óleo.

4.5.4. Identificação das Palhetas

Para identificação das palhetas, utilizou-se a ferramenta de contagem de objectos. Inicialmente definiu-se uma região de interesse, uma área entre dois círculos concêntricos. A ferramenta realiza uma contagem de todos os objectos – conjuntos de pixéis brancos delimitados por pixéis pretos – dentro dessa ROI. Basta então aplicar uma filtragem dos objectos tendo em conta a área (número de pixéis) e comprimentos dos seus eixos maior e menor (largura). No caso de contagem das palhetas, os dois círculos concêntricos da ROI foram posicionados de modo a intersectar a parte inferior e superior destas, com alguma tolerância, para que, caso as palhetas estejam ligeiramente deslocadas, os círculos nunca deixem de as intersectar (a verde, na figura 4.12). Isto faz com que os objectos correspondentes às palhetas possuam o mesmo comprimento ao longo do eixo maior, sensivelmente, tornando-os mais fáceis de filtrar.

Quanto à filtragem da área e eixo menor, para apenas identificar palhetas, a situação é mais delicada. A causa disto é o facto de as palhetas pretas não serem uniformes: ocasionalmente, palhetas pretas, além de terem os seus lados maiores foscos, também serão bicolores nos topos, em vez de polidas, como é o caso geral. Na figura 4.13 pode-se observar que, das palhetas

em posição vertical, a inferior é bicolor. Das restantes, a superior é uma palheta polida e a do meio é “escura”, mas de topos polidos. Aos olhos do programa de análise, estas palhetas tomam a forma de objectos com uma área cerca de metade da área de uma palheta uniforme no topo. Também o comprimento do eixo menor é afectado. É necessário diminuir os valores mínimos de filtragem, aumentando a gama de valores. Isto levanta incertezas na filtragem, assim como a margem de erro. Um pré-processamento eficiente é, portanto, a chave que garante que a análise de palhetas é bem realizada. Neste caso específico, uma binarização suave é suficiente para garantir que, além de as palhetas estarem perfeitamente separadas de qualquer outro objecto (como, por exemplo, o topo polido do rotor), certos reflexos (maioritariamente existentes do lado direito da figura 4.12), não são erroneamente tomados como palhetas pelo programa.

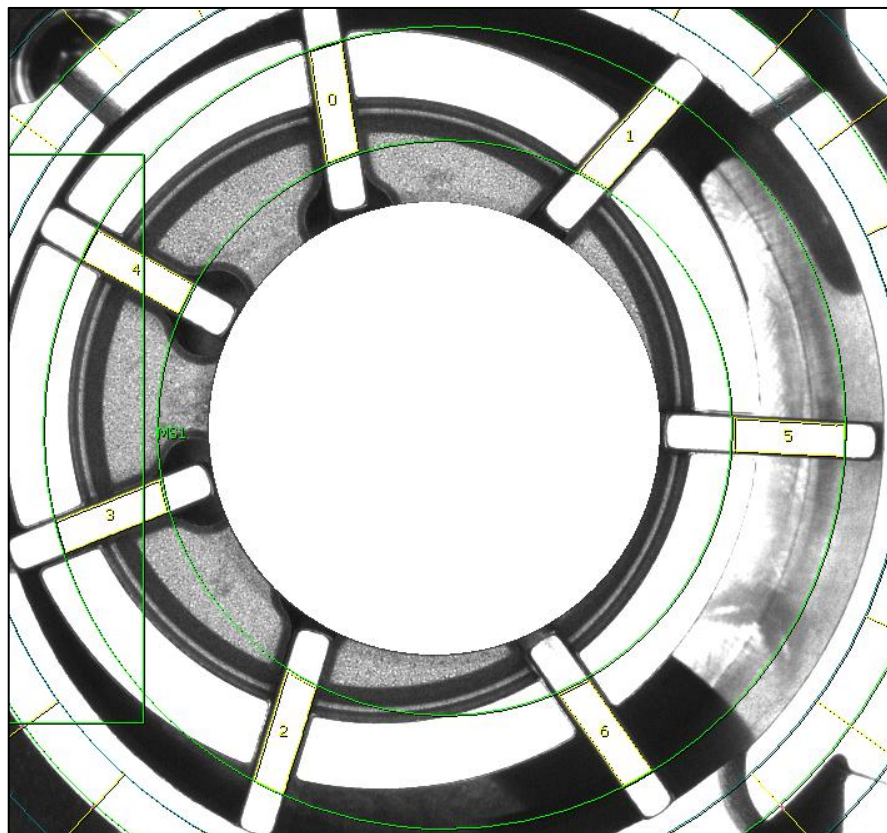


Figura 4.12: ROI de identificação de palhetas, com sete identificadas.

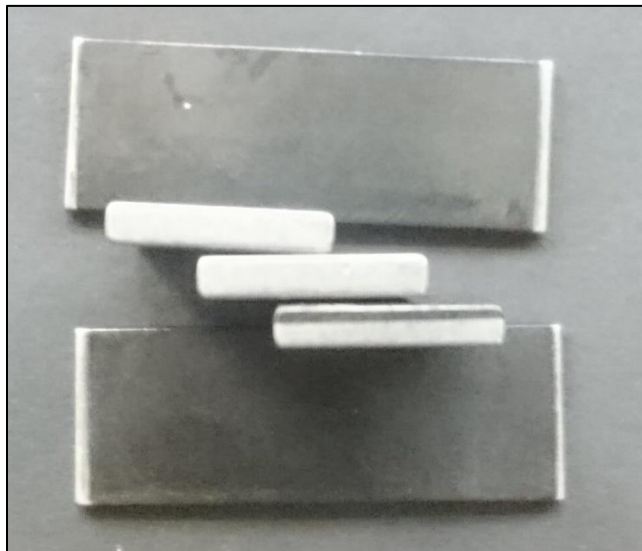


Figura 4.13: Topos de uma selecção de palhetas.

4.5.5. Identificação do Segmento

O segmento colocado entre o estator e o corpo da bomba de óleo é constituído por uma parte metálica, ligeiramente polida e de formato paralelepípedo e por uma borracha fixa ao metal. É possível observar na figura 4.8 que a zona de contacto entre estes dois materiais de aspecto díspar causa o aparecimento de gradientes relativamente fortes. Pode-se então verificar a existência destes “limites” nas sucessivas imagens retiradas de bombas de óleo para se concluir que o segmento existe ou não.

Pode-se utilizar a ferramenta de “*match*” para atingir esse fim. No entanto, contrastando com a procura de uma região para o localizador, esta comparação entre a imagem padrão, que contém o segmento, e outras, não é feita contabilizando todos os pixéis da imagem, mas simplesmente os limites entre regiões de diferentes valores na imagem. Os limites mais fortes da imagem padrão encontram-se representados a azul na figura 4.14. Com o segmento em falta, apenas os limites exteriores entre o estator e corpo da bomba e o local onde o segmento está inserido corresponderiam à imagem padrão, resultando numa disparidade grande e, portanto, um resultado negativo quanto à existência do segmento.

Apresenta-se também, na figura 4.15, uma análise efectuada numa bomba de óleo, mostrando as regiões de interesse e os valores obtidos para cada uma delas. Os valores da anilha, estator e sua concentricidade são o seu diâmetro em pixéis, com o valor óptimo entre parêntesis. O valor das palhetas é a sua contagem, com estas identificadas a amarelo. O segmento apresenta o seu valor de concordância entre a imagem padrão e a analisada, entre 0% e 100%. Neste caso, sendo a imagem padrão a ser analisada, o valor de concordância é máximo.

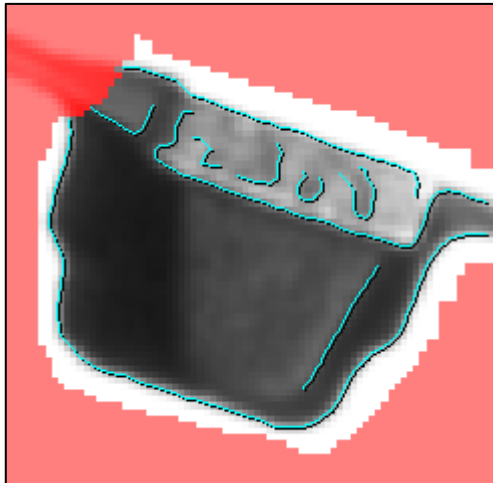


Figura 4.14: Máscara de procura do segmento, com limites identificados.

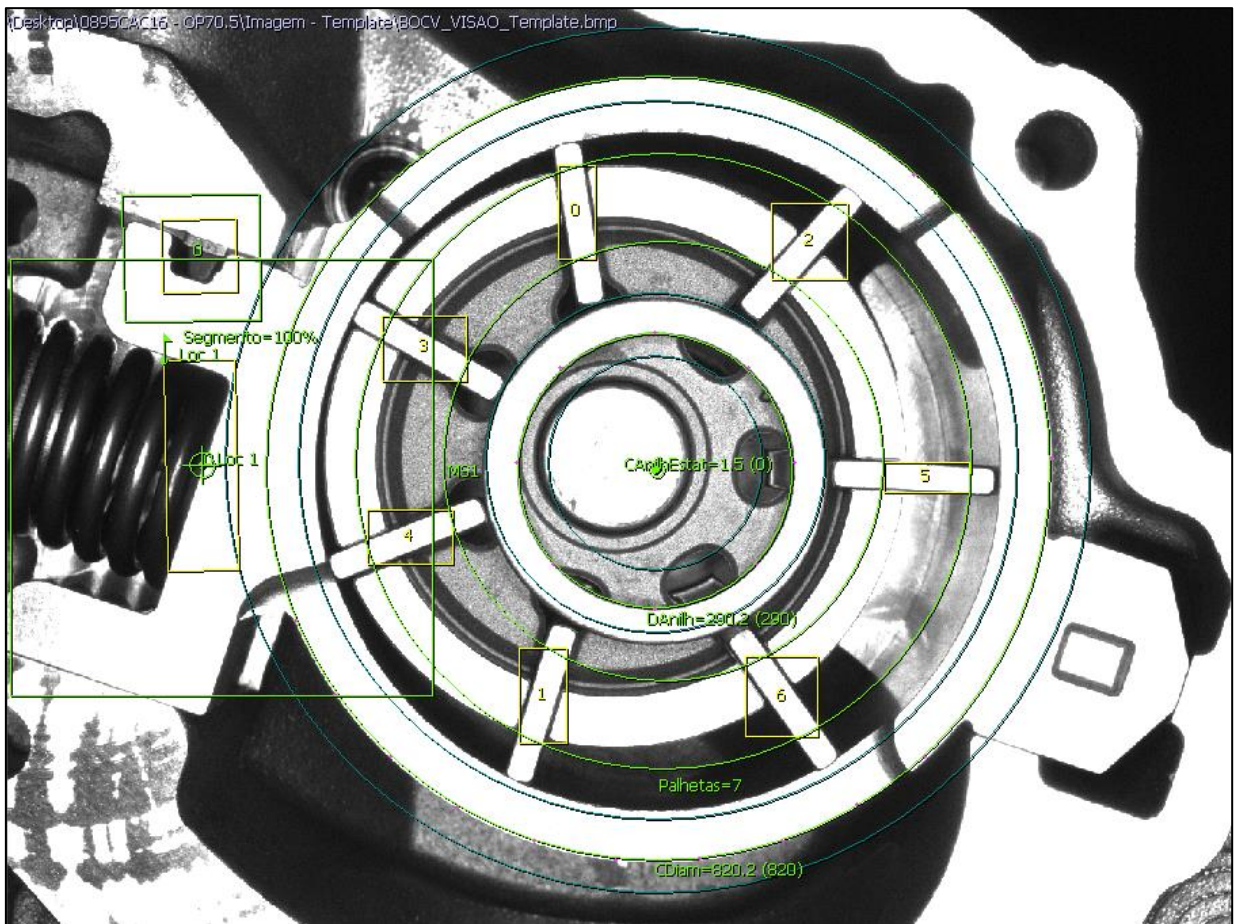


Figura 4.15: Análise realizada a uma bomba de óleo.

4.5.6. *Scripting*, Comunicação e Registos

Com a parametrização de identificação da anilha, palhetas e segmento concluída, é necessário comunicar com o autómato para receber o sinal de início de análise (“*trigger*”) e enviar os resultados dessa mesma análise, além de criar registos das análises.

Para troca de informação entre a *smart camera* – que processa internamente as imagens obtidas – e o exterior existem dois métodos possíveis: entradas e saídas digitais, e através de cabo Ethernet. Relativamente às entradas e saídas digitais, a câmara BOA está limitada a três saídas e duas entradas. Quanto às saídas, duas delas são configuráveis, com uma dedicada a ligar e desligar iluminação que pode ser conectada directamente à câmara (não utilizada). Para as entradas, uma é configurável, com a segunda dedicada à recepção de *triggers*. Ora, se as entradas são mais que suficientes, pois só se necessita da destinada ao envio de sinal de início de análise, as saídas são muito limitadas. É necessário enviar quatro sinais – um por resultados e um quarto de aviso de recepção (*acknowledgement*) – sendo que apenas se tem duas saídas. A solução é utilizar o cabo Ethernet, ligado entre câmara e autómato, para comunicação. Sendo que se trata de um PLC Siemens, a opção existente é a utilização de PROFINET. Com PROFINET, só se está limitado ao número de endereços (tanto entradas como saídas) atribuídos à câmara, durante a configuração dos dispositivos e módulos periféricos no programa do autómato. Como tal, a informação que é necessário transmitir entre autómato e câmara BOA podem ser transmitidos sem grandes problemas. Existe apenas um senão: enquanto que, utilizando I/O digitais, é possível enviar um sinal de *trigger* e a inspecção da bomba de óleo começar imediatamente, com PROFINET é necessário estar à escuta na rede pela recepção do sinal.

Para interagir com este array de valores e os resultados, recorreu-se às possibilidades de *scripting* oferecidas pelo software. Desenvolveram-se três scripts para comunicar com o autómato e criar registos em ficheiros de texto dos resultados. O primeiro, chamado a cada inicialização do programa (quando a câmara é ligada, por exemplo), limita-se a inicializar as diferentes variáveis, incluindo atribuir valores nulos à informação transmitida por PROFINET. O segundo, chamado ciclicamente a cada 10 ms, escuta a comunicação, à espera da recepção de um *trigger* e iniciando a análise de uma bomba caso o receba, ou reiniciando as variáveis caso o sinal se torne nulo. A existência de uma variável “cnt” garante que é apenas feita uma análise entre a recepção de um sinal de *trigger* e novo sinal de inspecção. O terceiro é executado após cada processamento e análise de imagens, e destina-se a enviar os resultados obtidos para o autómato, e criar registos. Para ser possível associar um registo a uma bomba de óleo específica, faz-se a leitura do código da bomba a ser analisada, enviado pelo autómato (que por sua vez o leu a partir da *tag* RFID contida na palete que contém toda a informação de produção relativa à bomba). De seguida associam-se os resultados à bomba e cria-se, ou edita-se, um documento de texto, alojado no computador do servidor da linha. A cada novo dia é criado um novo documento de texto. Finalmente, envia-se, dentro do *script* de pós-processamento, o *byte* de *acknowledgement*, para informar o PLC que os resultados já lhe foram enviados. É importante mencionar que os resultados enviados ao autómato podem tomar três valores: “0”, se a memória

não foi alterada (o que é equivalente, do lado do autómato, a não ter recebido esse resultado específico), “1” caso a análise tenha sido “má” (não identificou as setes palhetas, por exemplo) e “2”, se a análise tiver sido boa. Optou-se por estes valores, em detrimento de simples “0” e “1” para haver distinção entre um mau resultado e uma falha de comunicação. Nas figuras 4.16, 4.17 e 4.18 apresentam-se fluxogramas dos *scripts* anteriormente descritos.

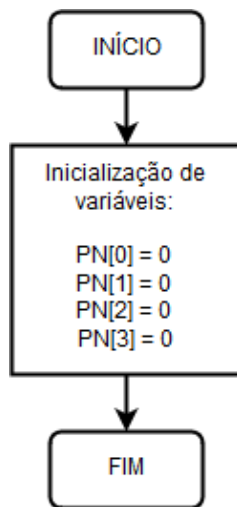


Figura 4.16: Inicialização da solução.

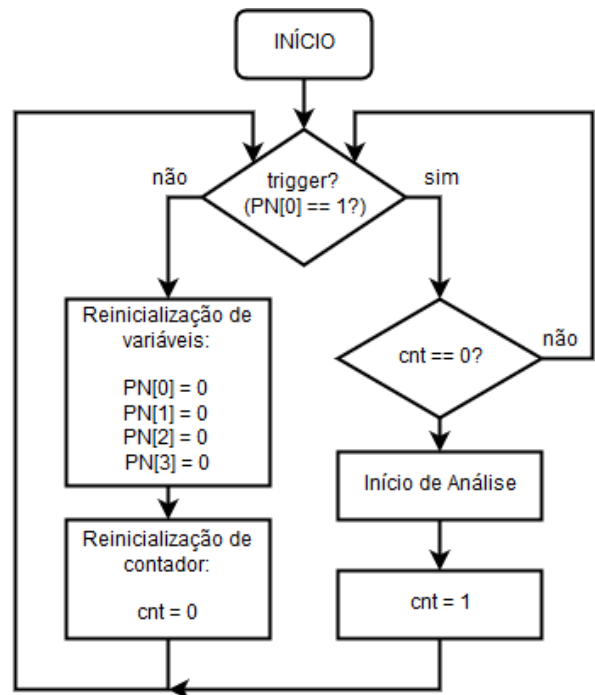


Figura 4.17: Script executado a cada 10 ms.

Além do registo de resultados, foi também implementado um registo de imagens provenientes de análises “más”. As imagens más servem não apenas para manter um registo visual das bombas em que componentes estavam mal posicionados ou ausentes, mas também para, em caso de o sistema realizar uma má inspecção (um falso positivo, por exemplo), a imagem pode ser analisada para despiste de erros de parametrização. Foram desenvolvidos dois métodos para guardar estes registos num computador na rede, neste caso o computador onde está alojado o servidor: o primeiro, através de *scripting*, inserido no *script* de pós-processamento, e o segundo, através de um método interno oferecido pelo iNspect Express. Infelizmente, o primeiro método, que é bastante mais flexível, permitindo a edição dos nomes das imagens de modo a associá-las a uma bomba específica, e que permite o registo tanto das imagens originais captadas como as imagens captadas com as regiões de interesse e análises efectuadas desenhadas apenas funciona com um cliente Windows XP, sendo que o sistema operativo do computador do servidor é Windows 7. Foi necessário, portanto, utilizar o segundo método, que atribui nomes correspondentes ao número da análise feita (por exemplo, “img31.bmp” para a imagem correspondente à trigésima primeira análise realizada com resultado “mau”). Este segundo método também só é capaz de exportar as imagens originais ou imagens com as análises feitas, mas não

ambas. Preferiu-se a exportação das imagens originais, para poderem ser subsequentemente importadas e analisadas, se necessário.

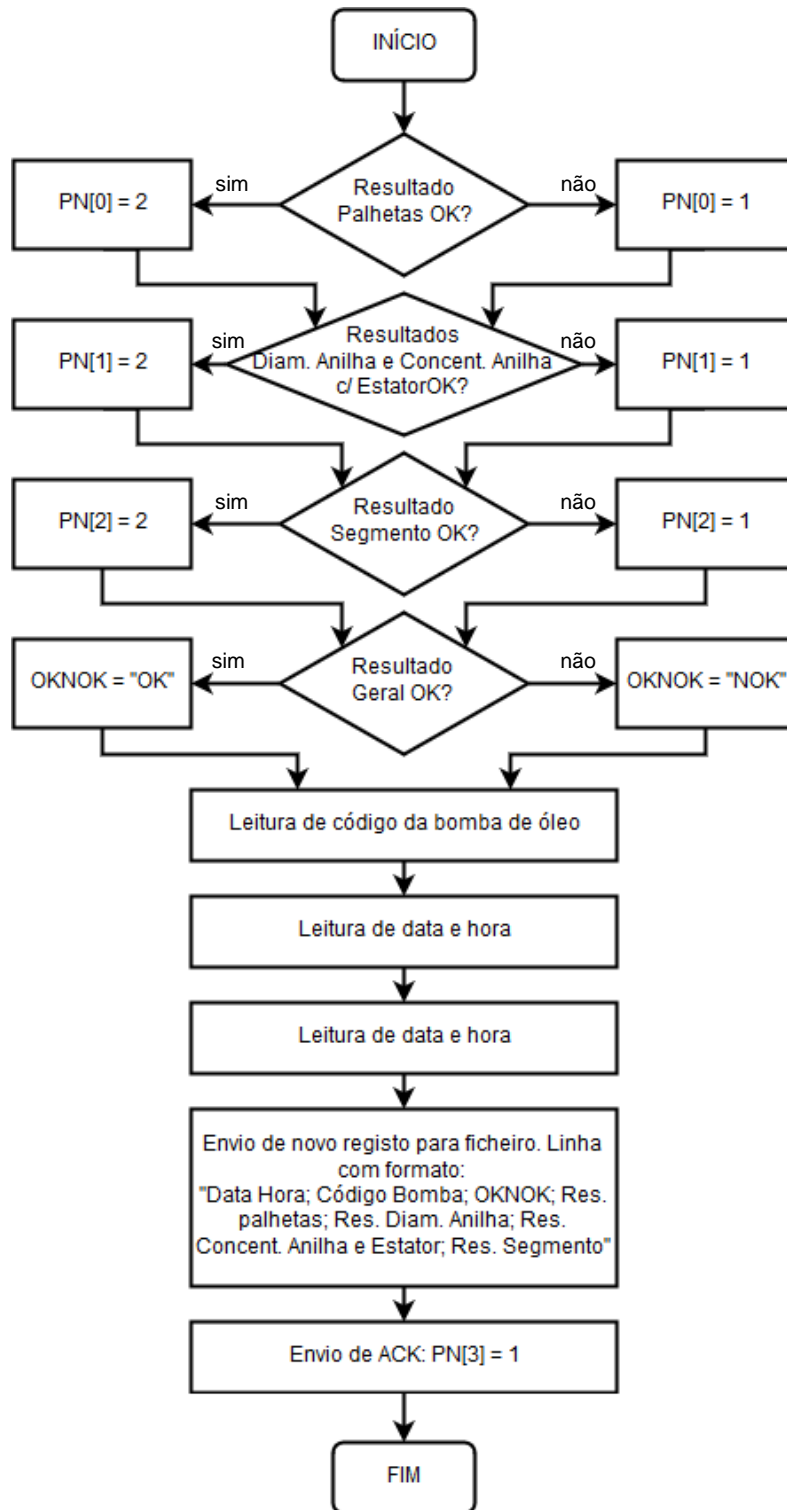


Figura 4.18: Script executado após realizada cada análise.

4.6. Programa do Autómato

O segundo programa desenvolvido para o sistema de visão consiste numa secção do programa geral do autómato do posto de trabalho OP70. A programação geral deste posto ficou a cargo de um colega de trabalho, que deixou o programa pronto para a inserção da interacção entre este e o sistema de visão: no ciclo automático, após subida do cilindro de bloqueio da paleta no posto OP70.5 e descida dos painéis laterais, é activada uma memória. Como tal, foi desenvolvido um ciclo dentro de uma função (FC), também ele automático, executado passo a passo, de envio de ordens e recepção de dados para e da câmara. Quando a memória descrita anteriormente se torna verdadeira (“1”), é iniciado o ciclo de comunicação com a câmara. Este, inicialmente, envia um sinal de *trigger* para o primeiro endereço de saída na rede PROFINET correspondente à câmara e o código da bomba a analisar, ficando a aguardar por uma resposta (ACK). A câmara, como dito anteriormente, após receber o *trigger*, realiza a análise, colocando as variáveis correspondentes aos resultados a “1” ou a “2”, dependendo do estado final da análise, enviando depois o sinal de *acknowledgement*. O autómato, mal receba esse sinal vindo da rede, lê os endereços correspondentes aos resultados, armazenando-os num *data block* (DB) – um objecto de armazenamento de dados – para futuro uso, e dá a análise como concluída. Ao ser dada a análise como concluída, o *trigger* é reinicializado, sendo colocado a “0”. A câmara, ao receber a informação de sinal de inspecção nulo, reinicializa as suas variáveis, ficando pronta a iniciar nova análise. A memória de “análise concluída” é utilizada depois pelo ciclo automático principal para continuar o seu seguimento, subindo os painéis laterais e descendo o cilindro bloqueador de paleta, fazendo com que a paleta com a bomba avance. A memória vinda do ciclo principal do programa do posto de trabalho de início de análise, não só sinaliza o envio do *trigger*, mas também causa a reinicialização das memórias relativas aos resultados, assim como a memória de trabalho finalizado.

No caso de haver um erro de comunicação, ou algo mais grave, como um cabo cortado, o autómato nunca irá receber o sinal de ACK. Para o programa do autómato não ficar infinitamente à espera de um sinal vindo da câmara, parando o ciclo automático do posto de trabalho inteiro, implementou-se um intervalo de tempo em que o PLC aguarda pelo sinal. Findo o tempo, a análise é dada como concluída, é activada uma memória de “*time-out*” e todos os resultados são dados como “maus”.

Na figura 4.19 apresenta-se um diagrama de actividade do programa do autómato.

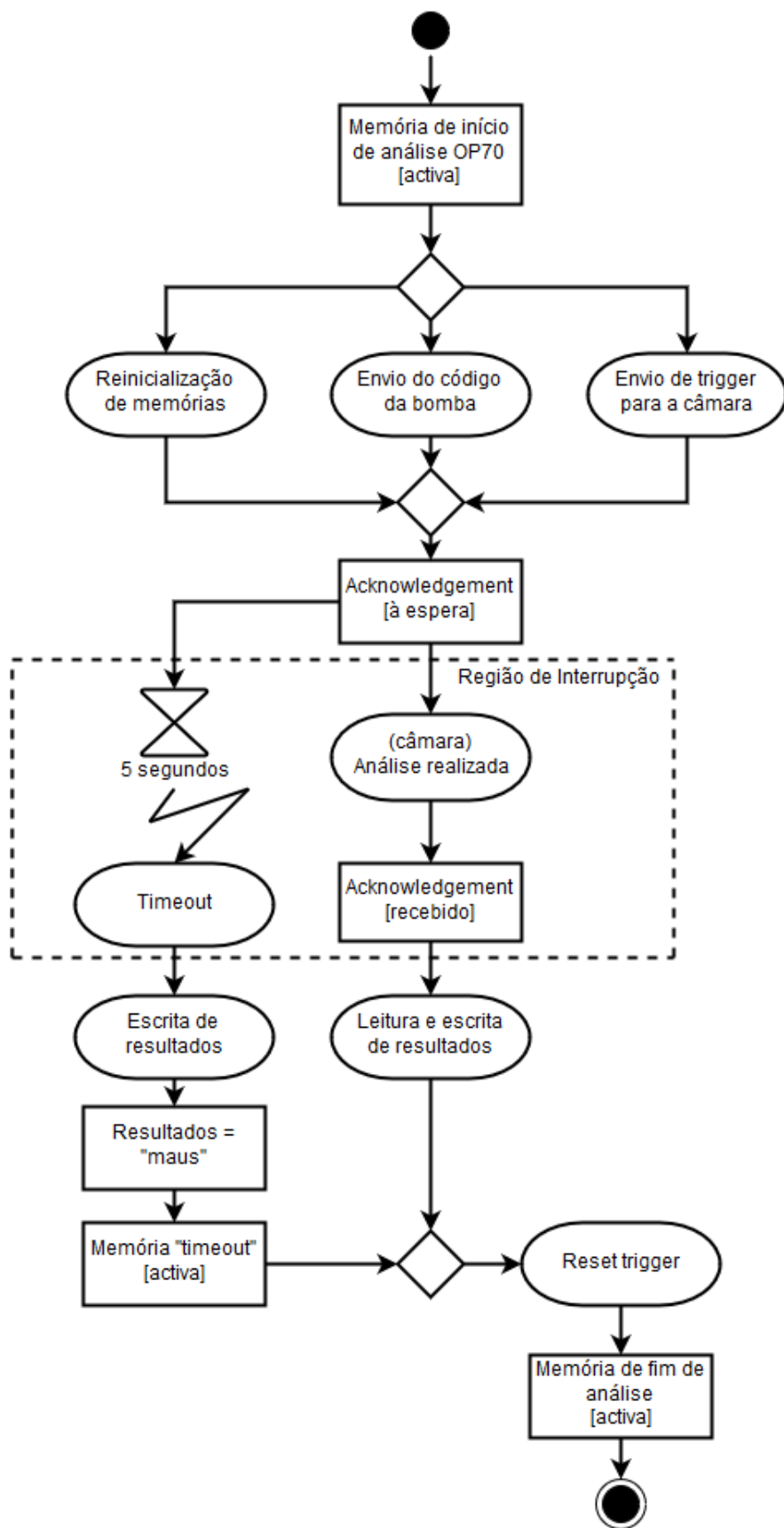


Figura 4.19: Diagrama de actividade do programa de PLC do sistema de visão.

4.7. Segundo Sistema de Visão

Após a realização do sistema de visão na linha de montagem, surgiu a oportunidade de desenvolver um sistema similar para uma linha de montagem de bombas de óleo, já em funcionamento contínuo e totalmente implementada (que produz exactamente os mesmos três modelos de bombas de óleo de caudal variável).

Este sistema de visão utiliza o mesmo princípio mecânico, com um cilindro elevador e bloqueador da paleta com a bomba e uma caixa negra de fundo aberto que aloja o sistema de visão. Também o programa de autómato é muito semelhante, tendo apenas alterações para o tornar capaz de interagir com o ciclo principal automático da linha de montagem, alterações essas realizadas por um colega de trabalho, e totalmente transparentes para a análise em si da bomba de óleo, fora o envio do código da bomba. O programa de análise foi adaptado, tendo sido modificadas as posições das ROI de análise de cada elemento, com alterações nos valores de binarização utilizados no pré-processamento e adaptação dos parâmetros de análise (por exemplo, a área em pixéis das palhetas, ou o diâmetro a procurar da anilha).

A câmara neste sistema de visão está mais distanciada do topo da bomba, fazendo com que os pixéis de interesse na imagem sejam em menor número (a zona de interesse, relativa à imagem completa, é mais pequena). Também o tamanho da abertura inferior da caixa é muito menor, comparativamente ao do sistema na linha de montagem nova, com a agravante de a caixa conter cilindros (para execução de operações de montagem não relevantes para o sistema de visão), ocupando espaço e dificultando o posicionamento do sistema de iluminação de modo a que este seja eficaz na iluminação da superfície da bomba de modo uniforme. Além de um anel de LED vermelhos similar ao utilizado no posto de visão anterior, utilizaram-se dois focos de luz branca difusa, para tentar uniformizar a luz na imagem captada. Nas figuras 4.20 e 4.21 pode-se ver um exemplo de imagem captada e imagem analisada, respectivamente, com uma análise “má” devido à falta de anilha, na segunda imagem. É possível observar nestas imagens a forma não totalmente uniforme com que o sistema de visão ilumina a bomba, principalmente na anilha e nas paredes do rotor. Daí surgiu a necessidade de realizar pré-processamentos mais robustos, mas mais pesados em tempo de processamento, nomeadamente para a análise das palhetas: contrastando com o sistema do OP705, utilizou-se não só uma binarização, mas antes desta uma erosão, para impedir que, ao binarizar, as paredes das palhetas se “unissessem” às paredes do rotor, dando origem a um único objecto, que obviamente cairia fora das tolerâncias definidas para as palhetas.

Quanto aos registos, foram inalterados, excluindo a eliminação da leitura do código da bomba, à semelhança da mudança realizada no programa do autómato, já que esta linha de montagem não efectua essa gestão. O computador de destino, em contraste com o que acontece na outra linha, é um mini PC localizado no quadro eléctrico, que não aloja um servidor de gestão da produção.



Figura 4.20: Exemplo de imagem captada pelo segundo sistema de visão.

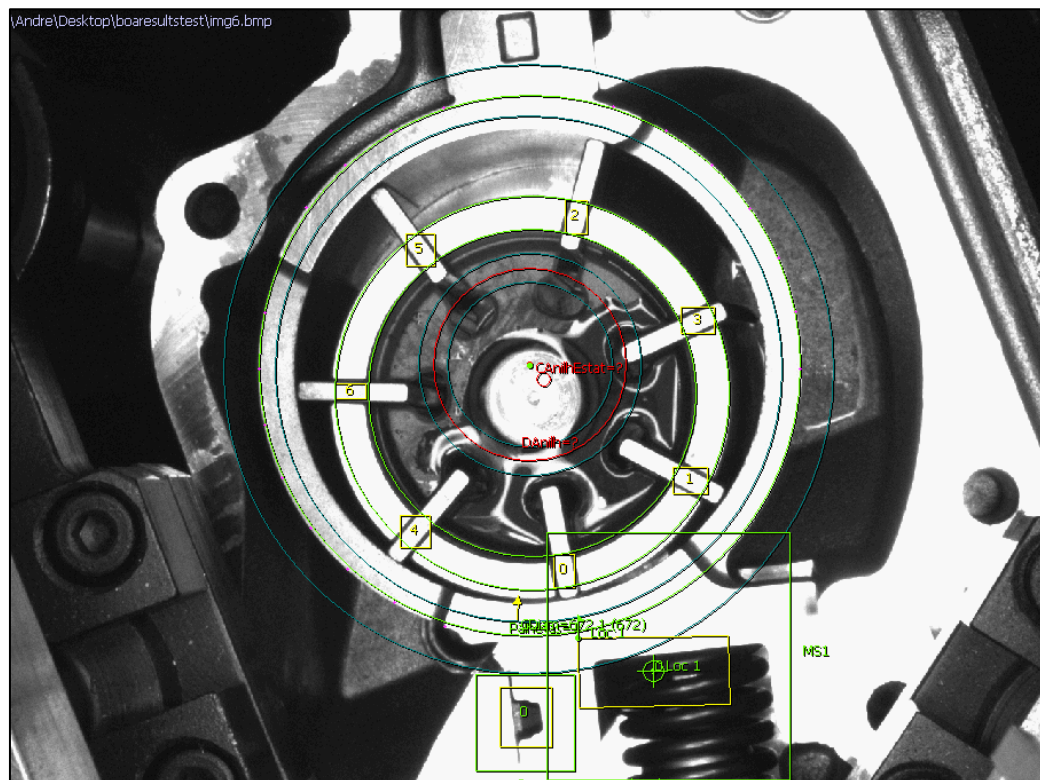


Figura 4.21: Exemplo de análise efectuada no segundo sistema de visão.

4.8. Resultados

Para validar o trabalho desenvolvido, foram realizadas várias análises para cada sistema de visão. Um entrave à realização de testes de fiabilidade foi o facto de a linha de montagem em desenvolvimento não estar a funcionar em modo contínuo, dificultando a análise de um número elevado de bombas de óleo. Já em relação ao segundo sistema de visão, as dificuldades prendem-se com a necessidade de reduzir ao máximo os tempos de paragem, seja durante alterações mecânicas ou de parametrização do programa e testes.

4.8.1. Sistema de Visão do OP70.5

Para testar e fiabilizar este sistema, foram introduzidas manualmente no posto OP70.5 várias bombas de diferentes modelos, em várias paletes para cada bomba. Realizaram-se várias análises para cada combinação diferente de bomba e palete – cinquenta para cada, exceptuando uma combinação incluindo a bomba M9T, de modo poder-se fiabilizar com maior segurança a existência de palhetas escuras – com e sem elementos presentes, variando a posição das palhetas (através da rotação do rotor), variando as palhetas introduzidas (de uma selecção de 25 palhetas diferentes) e variando a anilha (de quatro diferentes). No teste do sistema de visão relativamente à existência ou não de segmento, utilizou-se uma bomba de óleo de modelo R9M sem este elemento. As paletes não são universais, havendo dois modelos, um para bombas Hxx e outro para M9T e R9M. Este facto limitou o número de combinações possíveis. Estas combinações encontram-se compiladas na tabela 4.1. Também se compilou o tempo de execução da análise pelo programa da câmara de visão. Este tempo de processamento inclui não só a captura da imagem e o processamento da imagem em si, mas também o tempo necessário para realizar os registos de imagens e documento de texto, excluindo o tempo de comunicação entre autómato e câmara. Os resultados dos testes podem ser observados na tabela 4.2, na qual estes foram divididos em boas análises, com resultados “bons” e “maus”, e análises mal realizadas.

Combinação	Bomba	Palete
1	Hxx – 1	Hxx – 03
2	Hxx – 1	Hxx – 12
3	Hxx – 2	Hxx – 03
4	Hxx – 2	Hxx – 12
5	M9T – 1	R/M – 47
6	M9T – 1	R/M – 49
7	R9M – 1	R/M – 47
8	R9M – 1	R/M – 49

Tabela 4.1: Combinações de bombas e paletes usadas em testes.

Combinação	Result. “Bons”	Result. “Maus”	Análises c/ Erro	Tempo \bar{t} (ms)
1	45	5	0	170
2	40	10	0	176
3	40	10	0	179
4	38	12	0	174
5	82	18	0	157
6	28	22	0	161
7	0	50	0	165
8	0	50	0	172

Tabela 4.2: Resultados dos testes efectuados no sistema de visão do OP70.5.

Como se pode observar, o sistema é bastante robusto, não tendo ocorrido nenhuma análise errada, num total de 450. Os resultados “maus” incluem análises em que se retirou propositadamente a anilha, ou uma ou mais palhetas da bomba, excluindo as últimas duas combinações, em que o segmento foi omitido. No entanto, como era suposto, além do segmento, tanto as palhetas como a anilha foram bem identificadas nas cem análises. O tempo de análise é bastante curto, não se observando diferenças significativas entre diferentes combinações, e atingiu-se após optimizações tanto a nível de *scripts* como de pré-processamento – os utilizados e comentados anteriormente neste capítulo são os finais, utilizados neste teste – dado que originalmente, em análises iniciais, o tempo de processamento rondava os 500 ms.

Obviamente que estes testes, apesar de serem completos, não contemplam todas as possibilidades, pois não utilizam todas as paletes que circularão no sistema, e muito menos contemplam todas as bombas de óleo.

4.8.2. Sistema de Visão da Segunda Linha de Montagem

Neste sistema, a fiabilização envolveu permitir o normal funcionamento da linha de montagem, sendo contínua a passagem de bombas pelo sistema de visão, com a análise dos resultados do sistema feita recorrendo ao histórico de imagens e documentos de texto com registos das análises. Estes testes incluíram a passagem não controlada de paletes e bombas pelo sistema, pelo que apresentar uma tabela de combinações é totalmente irrelevante. Os resultados encontram-se apresentados na tabela 4.3, que representam os registos da análise de 4196 bombas de óleo. As análises indicadas como “combinação” foram análises onde o programa não conseguiu identificar dois ou mais elementos (por exemplo, palhetas e anilha ou anilha e segmento).

Resultados “Bons”		4092	Total (Total em %)
Resultados “Maus”	Palhetas	28	47 (1,12%)
	Anilha	12	
	Segmento	4	
	Combinação	3	
Más Análises	Palhetas	46	57 (1,36%)
	Anilha	0	
	Segmento	8	
	Combinação	3	
			4196 (100%)

Tabela 4.3: Resultados das análises no segundo sistema de visão.

Este segundo sistema de visão não é óptimo, tendo uma taxa de análises mal realizadas relativamente alto, acima de um ponto percentual. Isto deve-se ao sistema de iluminação ser incapaz de iluminar uniformemente a superfície da bomba de óleo a ser analisada. Isto pode levar (e leva, como se pode observar), em casos muito raros, ao aparecimento de reflexos nas superfícies que não sejam palhetas que podem ser contabilizados como sendo palhetas (também devido à tolerância grande nesta análise para poder detectar tanto palhetas totalmente polidas como bicolores no topo). Um acontecimento com uma ligeira maior ocorrência é o facto de tais reflexos fazerem com que, após a binarização no pré-processamento da procura de palhetas, uma destas se una a uma parte da parede do rotor, tornando impossível a sua detecção. Novamente, a iluminação menos que óptima pode induzir o sistema em erro, relativamente à detecção do segmento, fazendo com que o limite entre a parte metálica e de borracha deste seja pouco visível e indistinto da área em redor.

4.9. Conclusões

Um sistema de visão não é algo trivial de desenvolver e implementar. Além de ser necessário projectar e construir sistemas mecânicos que impeçam factores externos ao sistema, como é o caso de luz externa, de causar interferências, é necessário fazer uma boa selecção dos componentes a utilizar, sejam a câmara ou sensor para captar imagens, unidade de processamento e iluminação, de modo a criar o sistema mais adequado à resolução do problema. É necessário uma programação e parametrização cuidadosas para limitar as chances de se realizar más análises e, caso seja impossível, é sempre preferível que o sistema rejeite um produto bom que deixar seguir um produto defeituoso. No primeiro caso, pode-se repetir a análise, enquanto que no segundo caso, a menos que haja uma verificação posterior, irá invariavelmente causar problemas a jusante.

Os sistemas desenvolvidos são minimamente competentes na tarefa que lhes foi atribuída. No entanto, é necessário proceder a mais testes de fiabilização quanto ao sistema da linha de montagem em desenvolvimento, quando esta estiver a produzir bombas de óleo de caudal variável em modo contínuo, ou pelo menos num volume considerável. Já o segundo sistema de visão necessita de alterações ao nível da iluminação e/ou estruturais. Ou se utiliza luzes capazes de serem colocadas, dentro da caixa negra, ou até fora desta, que iluminem de forma uniforme a superfície da bomba de óleo, ou, de mais difícil implementação, já que a linha está a funcionar de forma contínua, alterar a caixa negra a nível estrutural, nomeadamente a abertura na sua base, falada anteriormente. Segundo o cliente para o qual o sistema de visão se destinou, este é mais preciso que o instalado originalmente. No entanto deveria melhorado, com vista à redução do número de análises mal efectuadas.

5. SISTEMA DE GESTÃO DE PRODUÇÃO

5.1. Introdução

A existência de um sistema de gestão de produção surgiu primordialmente da necessidade de garantir a rastreabilidade das bombas de óleo de caudal variável ao longo da produção destas na linha de montagem. As funcionalidades que foram necessárias desenvolver incluem:

- Leitura de receitas de produção de base de dados SQL e envio dessas informações de produção para a linha de montagem;
- Escrita em bases de dados SQL da produção efectuada relativa a cada bomba de óleo produzida e em processo de produção (actualização em tempo real dos dados de produção);
- Exportação e importação de históricos de produção;
- Apresentação gráfica e tabelada de estatísticas de produção;
- Gestão de reintrodução de bombas não concluídas e que tenham sido rejeitadas pela linha – por causa de defeitos, por exemplo – no sistema de montagem;

O desenvolvimento da estrutura do sistema, assim como do ambiente gráfico (a nível de organização e elementos a utilizar) ficou a cargo de um colega de equipa. Foi atribuído ao estagiário o desenvolvimento de variadas funções em ANSI C e sua integração no programa do servidor para possibilitar o bom funcionamento de algumas das funcionalidades acima mencionadas. O candidato também desenvolveu janelas gráficas para o programa.

Na figura 5.1 pode-se observar a janela principal do programa, com o estado dos postos de trabalho ao centro (com o exemplo do OP15 com alarmes activos), menu de janela à esquerda e produção à direita.

Este sistema foi desenvolvido de modo a correr num computador industrial, utilizando o sistema operativo Microsoft Windows 7.



Figura 5.1: Página principal do sistema de gestão e produção da linha de montagem.

5.2. Funções Desenvolvidas

Procurou-se criar funções simples, claras, modulares e extensíveis, de modo a construir uma biblioteca fácil de entender, analisar e adaptar a futuros projectos, caso seja necessário. Ao todo, desenvolveram-se catorze funções:

- `checkLogin()`: para verificar qual o utilizador actual do sistema;
- `exitRuntime()`: para sair do programa de gestão de produção;
- `fileDialog()`: para apresentar o diálogo de abertura ou armazenamento de ficheiros do Windows;
- `fileSearch()`: para procura de ficheiros em pastas. Utilizada em conjunto com a função “`pumpSearchImport()`”;
- `logInOut()`: para um utilizador poder entrar no programa utilizando uma conta, ou alternativamente sair da conta;
- `NOPtoOP()`: manipula uma *string* com o número do posto de trabalho, devolvendo o nome deste;
- `OPtoNOP()`: semelhante à função anterior, mas de transformação no nome em número do posto de trabalho;

- `pumpSearchImport()`: para procura de palavras-chave em ficheiros e importação da informação associada à palavra-chave para o programa (a importação propriamente dita é realizada pela função “`uaImportentries()`”);
- `uaEditID()`: edita tabelas de históricos de produção guardadas em ficheiros “.csv”, eliminando a coluna de “ID”;
- `uaExportEntries()`: exporta manual ou automaticamente históricos de produção da base de dados para registos em ficheiros “.csv”.
- `uaGetParts()`: contabiliza número de peças boas ou más produzidas, lendo a base de dados de produção efectuada;
- `uaImportEntries()`: importa informação de produção realizada de ficheiros “.csv” para a base de dados de produção;
- `updateProdGraf()`: actualiza os gráficos das estatísticas da produção semanal realizada;x
- `writeLog()`: escreve históricos de importação e exportação e de erros.

5.2.1. `checkLogin()`, `logInOut()` e `exitRuntime()`

A função `checkLogin()` e `logInOut()` destinam-se a permitir um controlo maior ou menor do programa por parte do utilizador, interagindo com a funcionalidade interna do WinCC de criação de contas de utilizador. A primeira (figura 5.2) compara um dado nome de utilizador com o nome do utilizador actual, permitindo ou não o acesso à interacção o objecto no ambiente gráfico a que está associada, dependendo do operador actual. A segunda (figura 5.3) limita-se a permitir a entrada ou saída de contas de utilizador, dependendo da *flag* passada como argumento. No caso de se efectuar um *login*, a função apresenta uma janela onde é possível introduzir nome de utilizador e palavra passe. A função `exitRuntime()` (figura 5.4) sai do programa do sistema de gestão de produção, mas não sem antes apresentar uma janela ao utilizador, para confirmação.

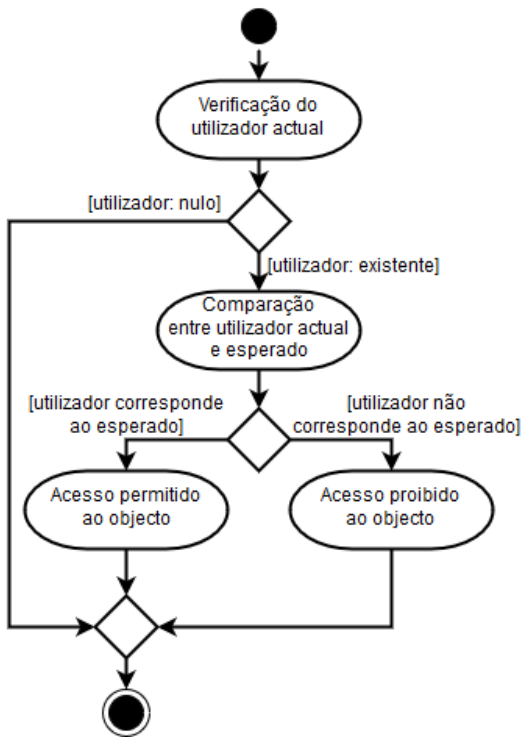


Figura 5.2: Diagrama de `checkLogin()`.

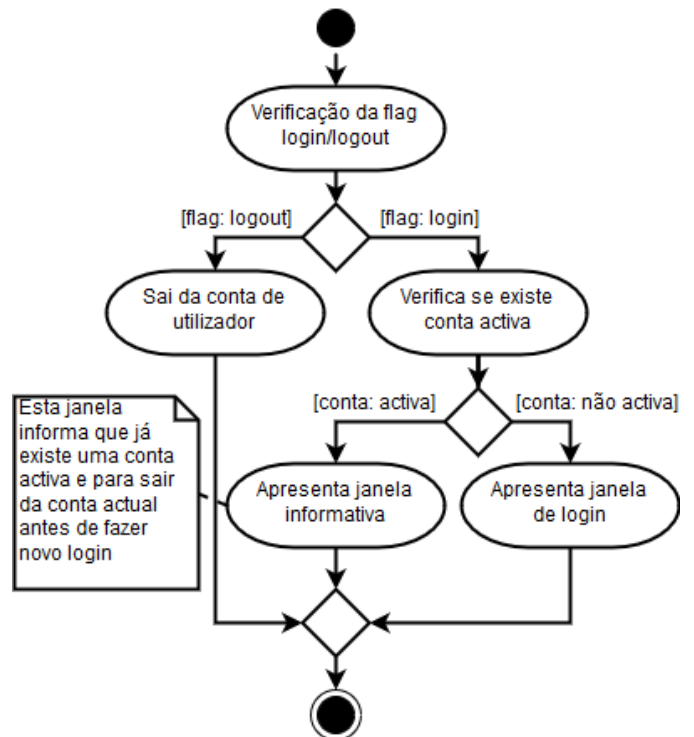


Figura 5.3: Diagrama de `logInOut()`.

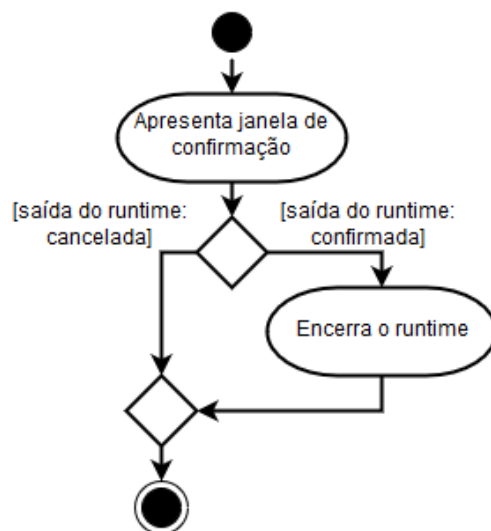


Figura 5.4: Diagrama de `exitRuntime()`.

5.2.2. OPtoNOP() e NOPtoOP()

Graficamente, o nome apresentado ao utilizador do sistema, relativamente a um posto de trabalho, toma a forma de “OP70_5” (para o OP70.5) ou “OP25”, por exemplo. No entanto, um nome com este formato é relativamente difícil de processar internamente pelo programa. Assim, desenvolveram-se duas funções – NOPtoOP() e OPtoNOP() – que realizam a conversão do nome do posto apresentado para um valor numérico, ou vice-versa: um nome “OP70_2” toma a forma de “702”, com um posto “OP60” a tomar o valor “600”, enquanto que um posto “OP160” é representado por “1600”. O nome ou número que se pretende converter é passado como argumento às funções. Estas funções são apresentadas nas figuras 5.5 e 5.6.

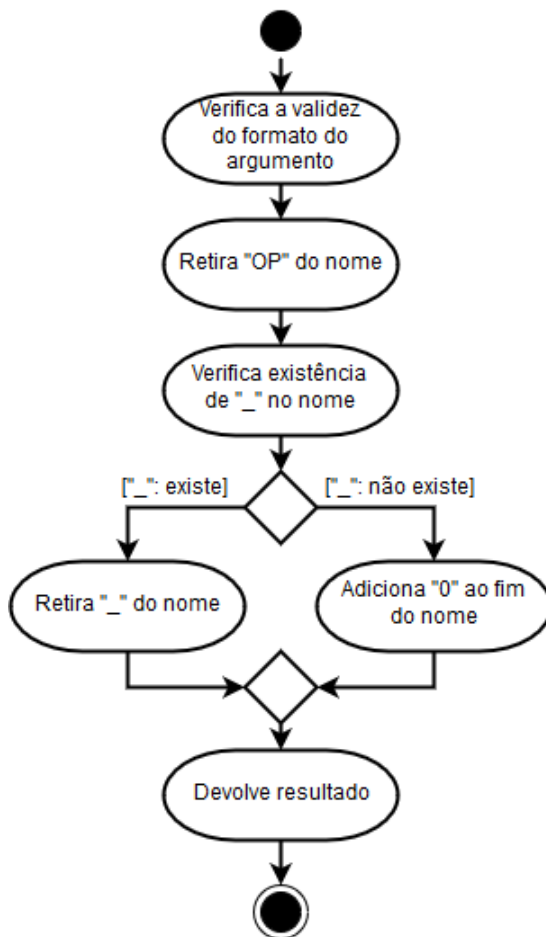


Figura 5.5: Diagrama de OPtoNOP().

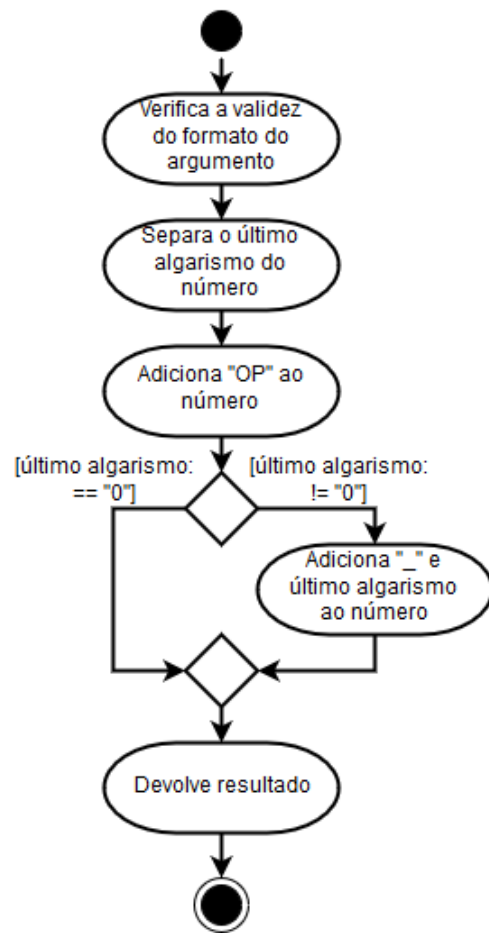


Figura 5.6: Diagrama de NOPtoOP().

5.2.3. openFileDialog()

Esta função apresenta uma caixa de diálogo que permite abrir ou guardar ficheiros, utilizando para isso a biblioteca dinâmica “comdlg.dll” (de diálogos comuns do sistema operativo). Estas janelas permitem a selecção de ficheiros ou a escrita dos nomes destes, devolvendo o resultado sob a forma do directório e nome do ficheiro, para posterior processamento. Na figura 5.7 apresenta-se a caixa de diálogo para abertura de ficheiros, e na figura 5.8 um diagrama de actividade da função.

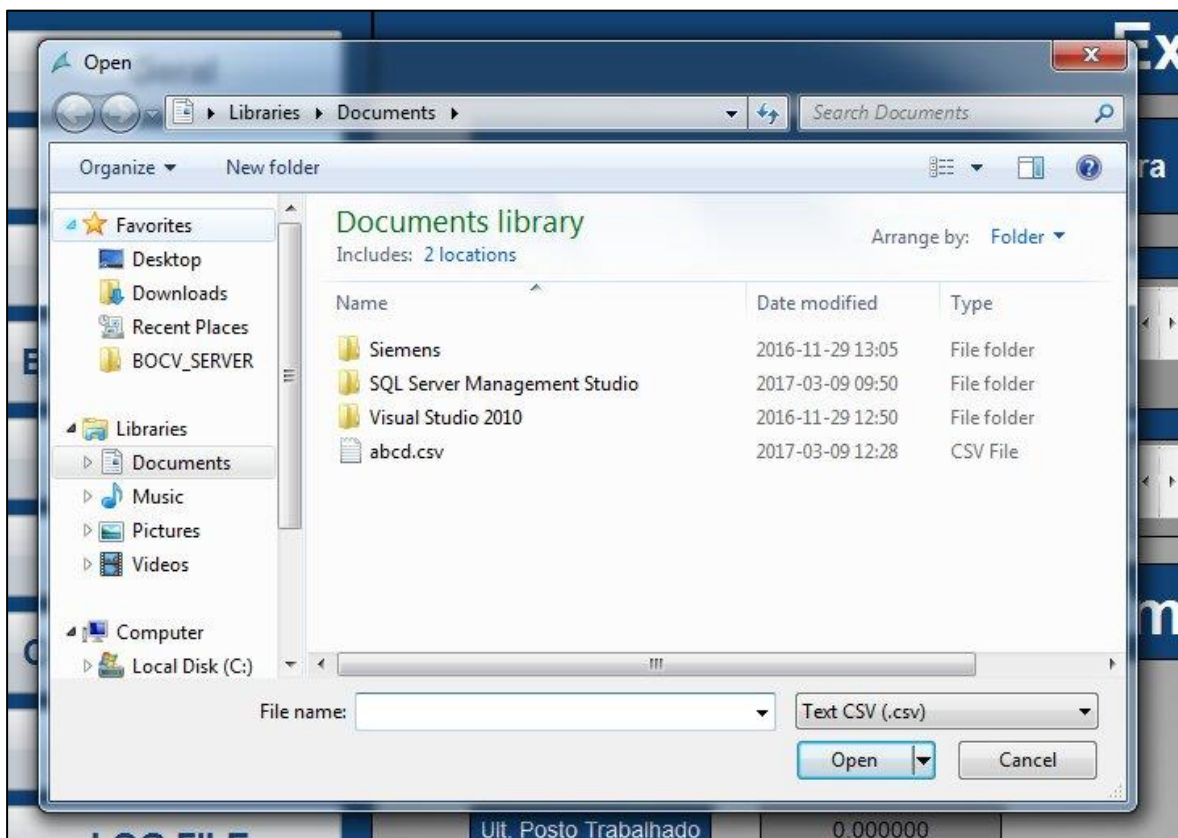


Figura 5.7: Caixa de diálogo de abertura de ficheiro.

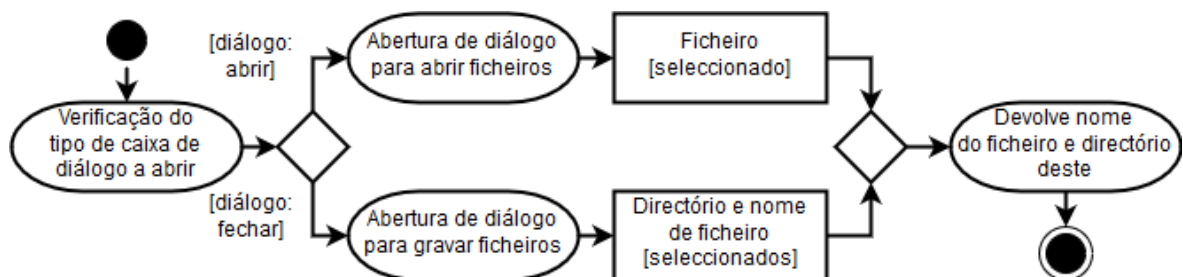


Figura 5.8: Diagrama de openFileDialog().

5.2.4. uaExportEntries() e uaImportEntries()

As funções `uaExportEntries()` e `uaImportEntries()` manipulam a base de dados onde é armazenada a informação de produção, exportando o seu conteúdo para ficheiros externos ou importando informação para a tabela desta.

De modo a não permitir que a base de dados contenha demasiadas entradas (sendo que é criada uma por bomba produzida), o que tornaria a manipulação dos seus dados lenta, criou-se a função de exportação. Esta, dependendo da *flag* de tipo de exportação passada como argumento, realiza a exportação automática ou manualmente. No caso de uma chamada da função para exportação automática, chamada automaticamente pelo programa de modo intervalado (a cada quinze dias, por exemplo), esta arquiva num ficheiro “.csv” todas as entradas da base de dados, excepto as mais recentes (por exemplo, todas as entradas excepto as dos últimos dez dias), eliminando-as de seguida da base de dados. Em modo manual, é necessário definir o intervalo temporal das entradas que se deseja exportar (figura 5.9, em cima), com a diferença adicional que a exportação manual não elimina entradas da base de dados.

Quanto à função de importação, é possível seleccionar um ficheiro – recorrendo à função “fileDialog”, que não faz parte da função em si – e deste importar todas as entradas neste contidas, caso possuam o mesmo formato da tabela de produção. A interacção entre a base de dados e o programa de supervisão e gestão é efectuada recorrendo a funções internas deste, que permitem seleccionar uma base de dados, filtrá-la, reorganizá-la, e, de um modo geral, manipular a informação nesta contida. As figuras 5.10 e 5.11 apresentam diagramas das duas funções.

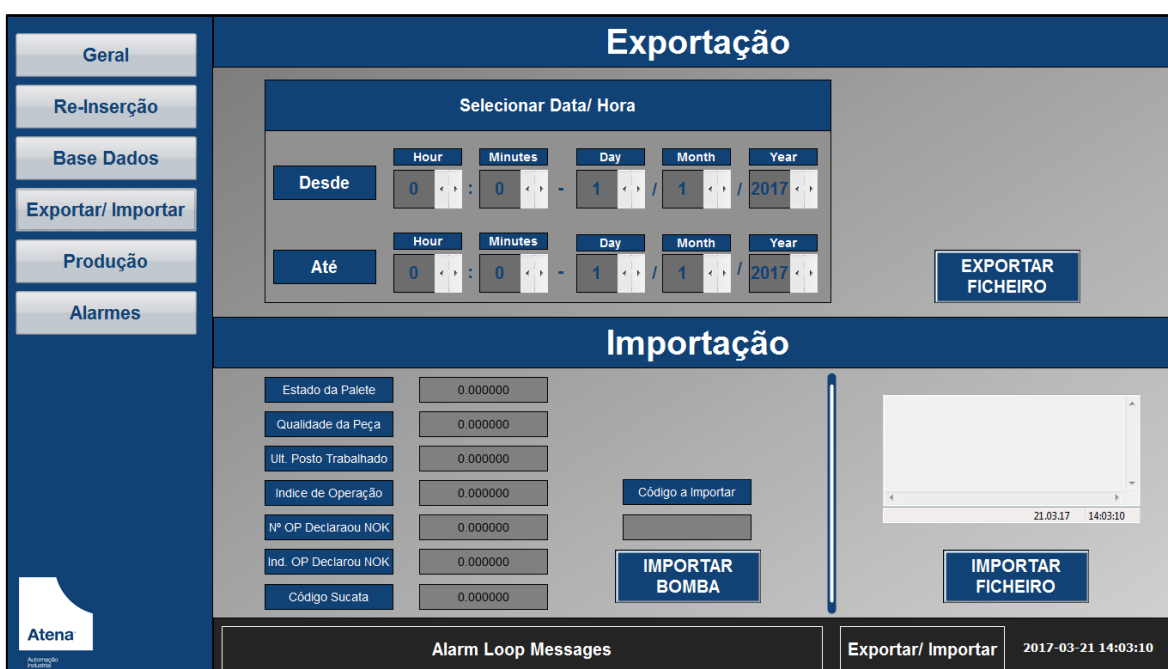


Figura 5.9: Janela de importação e exportação do sistema.

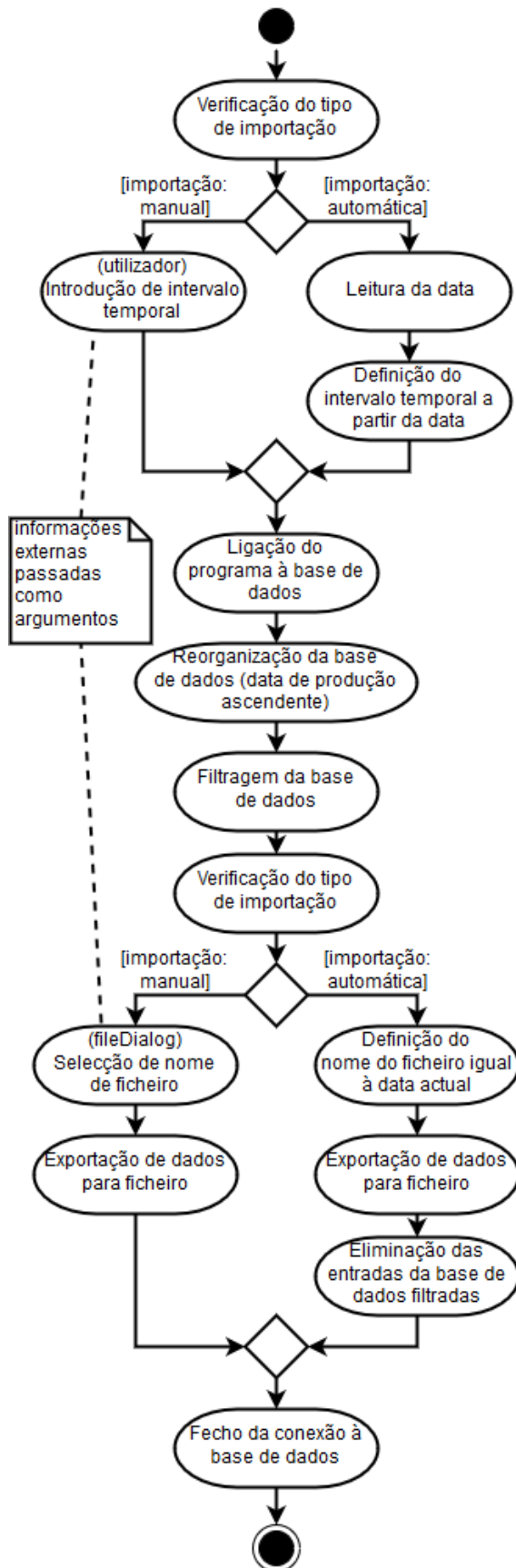


Figura 5.10: Diagrama de `uaExportEntries()`.

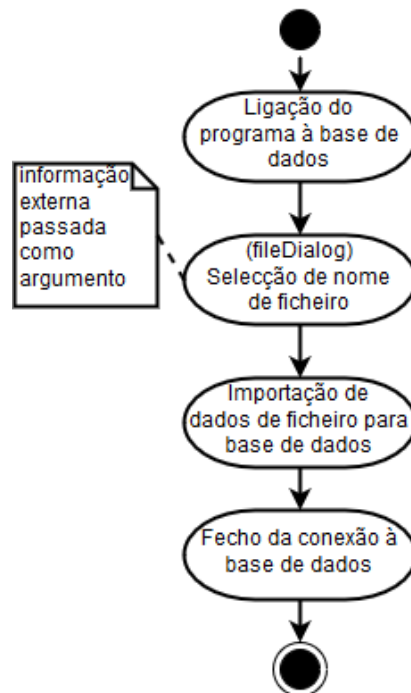


Figura 5.11: Diagrama de `uaImportEntries()`.

5.2.5. fileSearch(), pumpSearchImport() e uaEditID()

A importação de dados para a base de dados é feita em bloco, não sendo possível importar apenas uma entrada. Caso surja a necessidade de reintroduzir uma bomba de óleo na linha de montagem que tenha ficado tempo suficiente fora do sistema para que a exportação automática tenha arquivado a entrada correspondente, é necessário realizar uma procura dessa mesma entrada nos ficheiros do histórico, para depois então se importar a informação. A função `pumpSearchImport()` foi desenvolvida para proporcionar essa capacidade de pesquisa ao sistema. Dando o código identificativo da bomba de óleo (figura 5.9, em baixo) e o ficheiro onde procurar como argumentos, a função irá procurar neste, linha a linha, uma linha que contenha o código da bomba em questão. Caso encontre, armazena num ficheiro temporário o cabeçalho da informação contida no ficheiro seguido da linha contendo a informação da bomba de óleo, finalizando-se o processo chamando a função `uaImportEntries()` que importará todo o conteúdo desse ficheiro temporário. Antes da importação são também apresentados na janela de importação/exportação a informação da bomba de óleo a importar, para o utilizador confirmar os dados desta antes de realizar a importação. O método de pesquisa de texto em ficheiros é baseado na função “`fgetc()`” das bibliotecas *standard* de C. Esta função vai lendo o ficheiro carácter a carácter, até ser encontrado o final de linha. Nesse momento, a linha inteira é armazenada e analisada.

A função `fileSearch()` permite que a pesquisa seja realizada não só num ficheiro, mas em todos os ficheiros da pasta onde estão contidos os arquivos. Esta usa funções contidas na biblioteca dinâmica “`kernel32.dll`” do Windows (`FindFirstFileA()` e `FindNextFileA()`) para ser possível navegar no sistema de ficheiros do sistema operativo. A cada novo ficheiro “.csv” encontrado, é chamada a função `pumpSearchImport()` para realizar a sua pesquisa. Caso tenha sido encontrada a informação necessária antes de se ter percorrido todos os ficheiros (ou se se tenha encontrado os dados a meio de um ficheiro) todo o processo de pesquisa é abortado, já que não é necessário continuar a pesquisa até ao fim.

A primeira coluna da tabela de produção da base de dados é o identificativo da entrada (“ID”). Este valor (assim como o código de bomba) é único, não podendo existir duas entradas com o mesmo valor. No entanto, observou-se que, em alguns casos (falha de energia e reinicialização do sistema, por exemplo), o valor do ID, caso se tivesse arquivado entradas, era reinicializado. Isto oferece problemas, já que é possível ter um ficheiro com uma entrada de ID igual a “4” e uma entrada totalmente diferente no sistema com o mesmo valor. Para contornar o problema, desenvolveu-se a função `uaEditID()`, que, após, arquivar entradas num ficheiro (exportando a informação) edita essas mesmas entradas, dando um valor nulo (vazio) aos valores de ID das entradas. Ao importar essa informação, essa coluna será automaticamente preenchida com novos valores, mantendo-se o código da bomba para garantir que não existem entradas duplicadas. A função `uaEditID()` é uma reimplementação da função `pumpSearchImport()`, mas

em modo de edição de ficheiro em contraste com o modo de leitura da anterior. Estas funções encontram-se expostas nas figuras 5.12, 5.13 e 5.14.

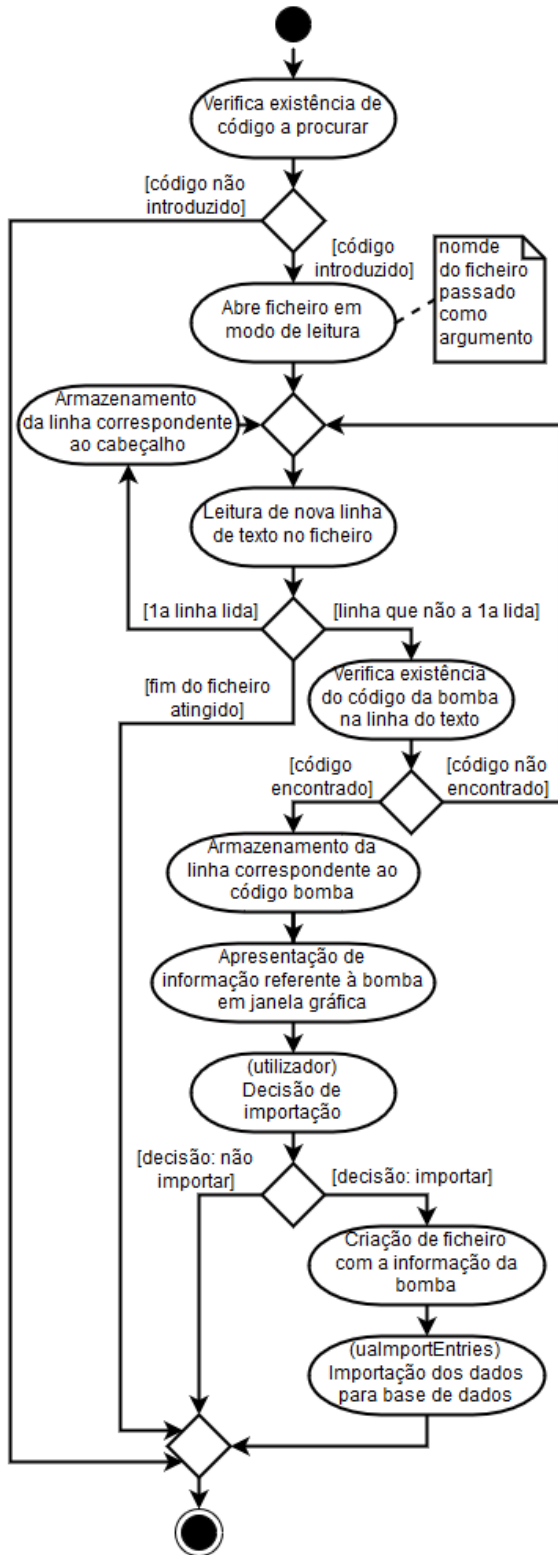


Figura 5.12: Diagrama de `pumpSearchImport()`.

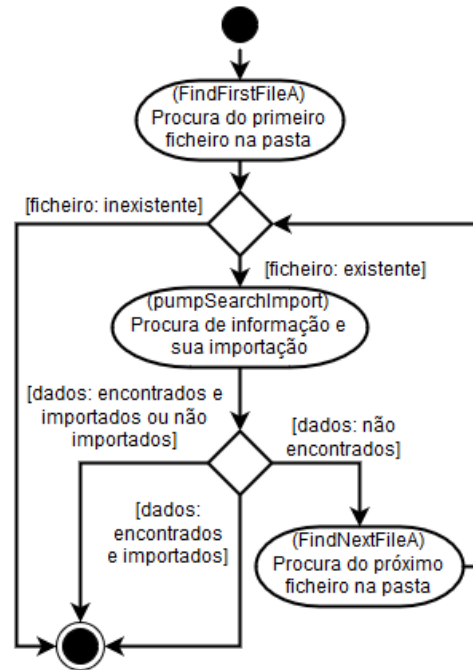


Figura 5.13: diagrama de `fileSearch()`.

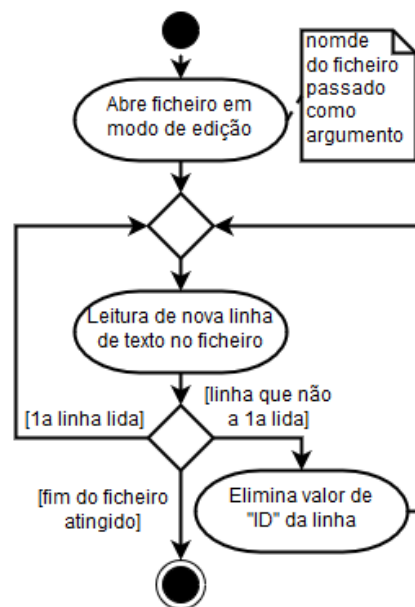


Figura 5.14: Diagrama de `uaEditID()`.

5.2.6. updateProdGraf() e uaGetParts()

Estas funções são destinadas à manipulação de propriedades de objectos gráficos na janela de estatísticas de produção semanal do sistema de gestão (figura 5.15), também ela desenvolvida pelo candidato. Esta janela apresenta a produção realizada no dia actual e sete dias anteriores a este (peças boas e peças más), sendo que a função updateProdGraf() actualiza o gráfico do dia “hoje” a cada novo acesso à página. A cada novo dia, toda a informação do gráfico mais recente é enviada para o gráfico seguinte (de “hoje” para “ontem”, por exemplo) e assim por diante, actualizando todos os gráficos, seguindo uma lógica FIFO – “*first in first out*”, ficando o gráfico do dia actual limpo para começar a ser preenchido com a produção do dia.

A função uaGetParts() é utilizada para fornecer à função anterior os dados necessários ao preenchimento dos gráficos: esta acede à base de dados da produção, retirando o número de bombas bem produzidas e a quantidade de bombas “más”.

Na figura 5.16 e 5.17 expõem-se os diagramas de ambas as funções.

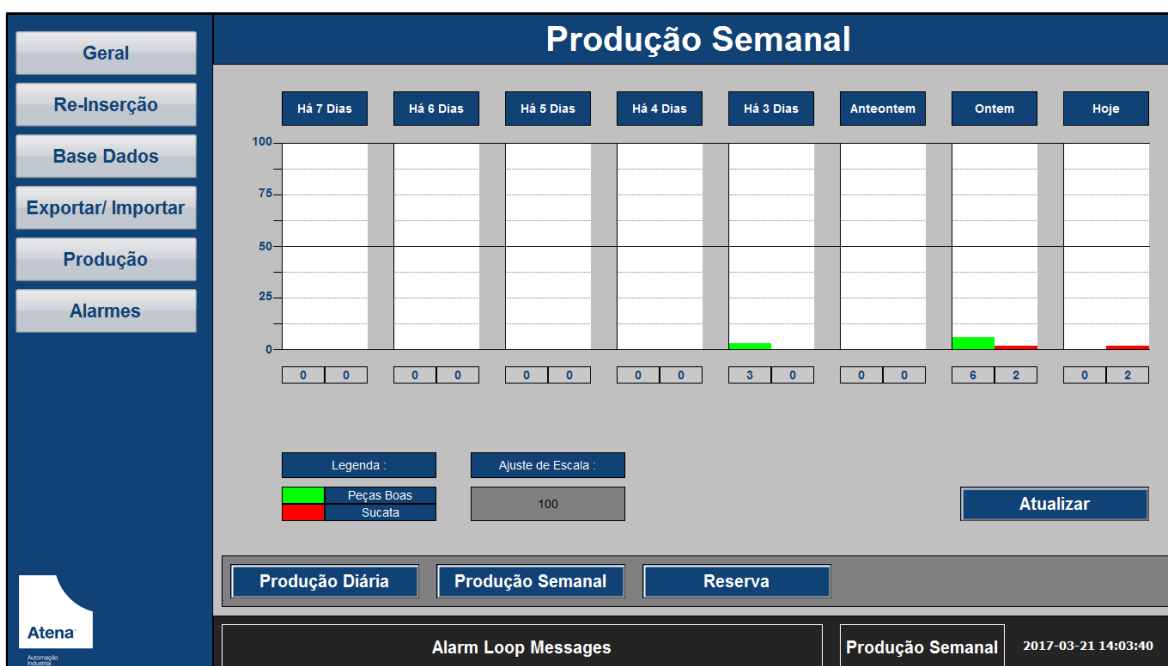


Figura 5.15: Janela de produção semanal.

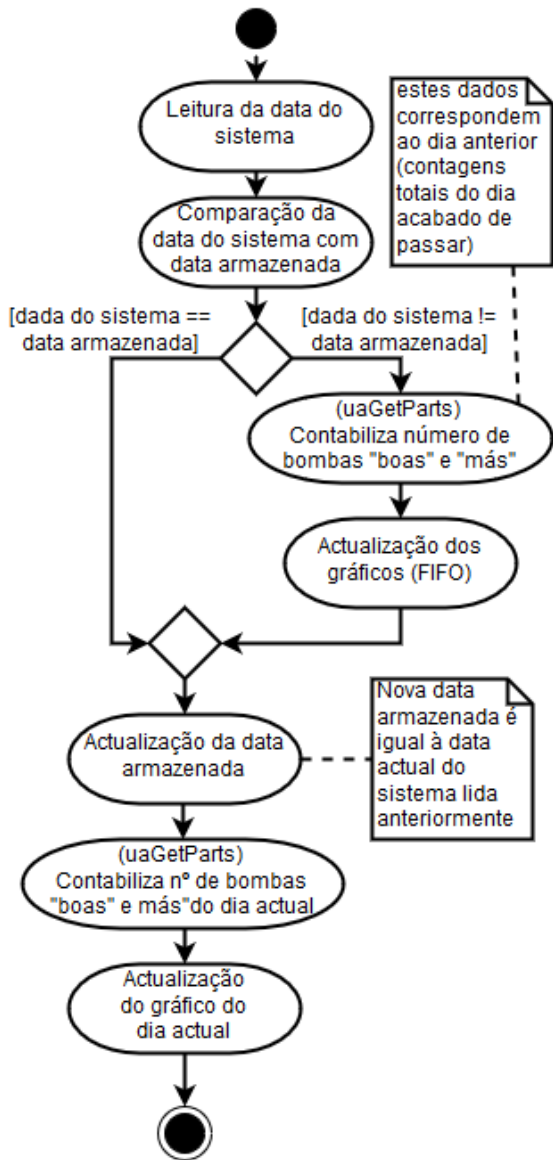


Figura 5.16: Diagrama de `updateProdGraf()`.

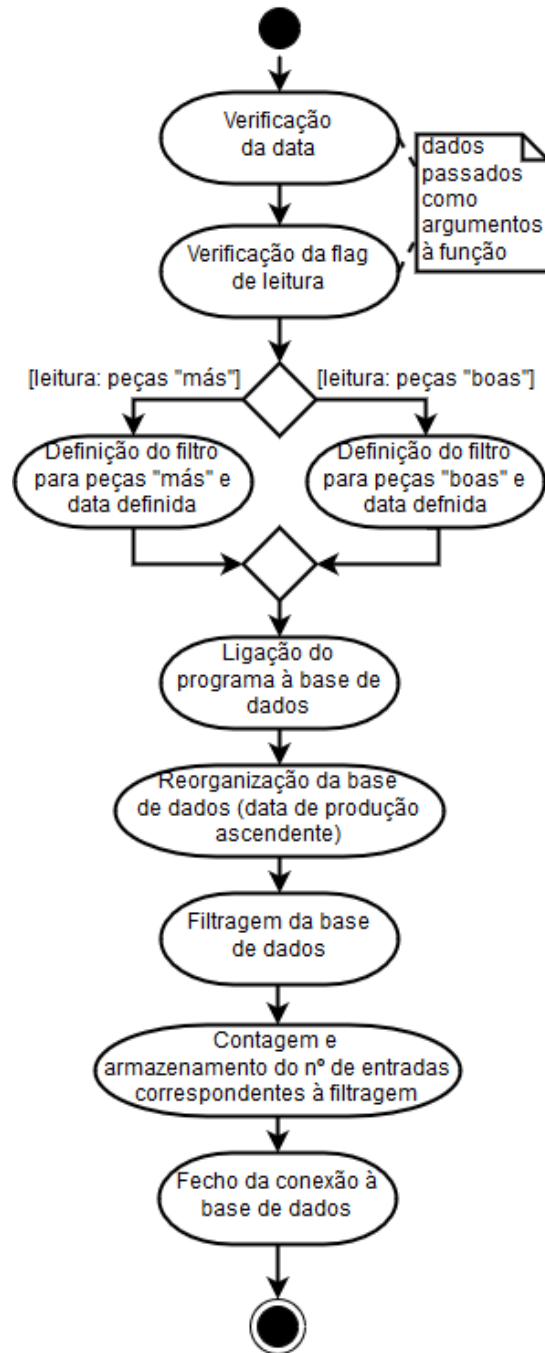


Figura 5.17: Diagrama de `uaGetParts()`.

5.2.7. writeLog()

Este é um sistema que é necessário encontrar-se em bom funcionamento vinte e quatro horas por dia, sete dias por semana. Como tal, é de extrema importância a existência de históricos de erros, para que estes permitam uma análise rápida dos possíveis problemas com o sistema e uma também rápida resolução destes. Estes históricos de erros são proporcionados pela função writeLog(), que regista o último código de erro criado pelo programa e o armazena num ficheiro de texto. Além desta funcionalidade, esta função também regista num histórico todos os momentos em que é realizada uma exportação de dados da base de dados de produção de bombas de óleo, automática ou manual, registando a data e hora de exportação, o directório e nome do ficheiro onde foram armazenados os registos e a quantidade de entradas arquivadas, permitindo saber se em algum momento foi realizada uma exportação de dados indevida. Um diagrama de actividade relativo à função encontra-se apresentado na figura 5.18.

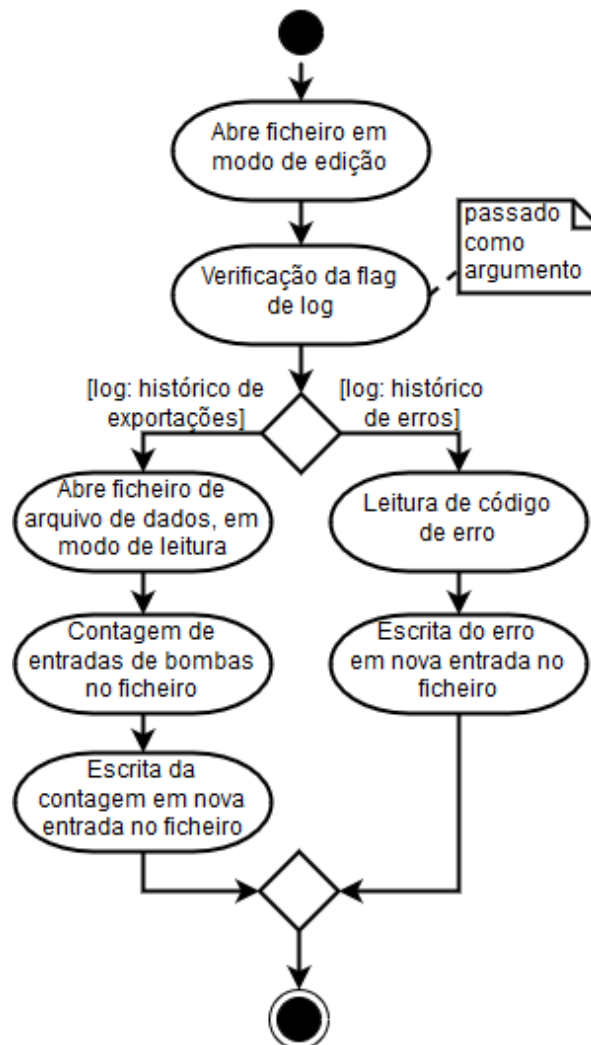


Figura 5.18: Diagrama de writeLog().

5.3. Resultados

As funções foram desenvolvidas com o propósito de ser possível associá-las e encadeá-las, ou reutilizá-las em futuros projectos. De modo geral, tentou-se que fossem independentes do sistema de supervisão e gestão. Isto revelou-se uma tarefa impossível em algumas funções, já que o seu funcionamento dependia obrigatoriamente da chamada de funções internas do sistema, ou o armazenamento de valores em variáveis globais do sistema, para futura interacção deste com elas. No entanto, é possível a sua utilização noutros sistemas de gestão que utilizem o *software* WinCC da Siemens, com pequenas alterações. A única função que se pode considerar realmente impossível de dissociar do sistema desenvolvido é a referente à manipulação de gráficos de produção semanais (`updateProdGraf()`), já que se trata de um caso bastante específico.

Relativamente ao desempenho destas funções, teve-se o cuidado de criar funções que utilizassem os métodos mais rápidos para realizar a tarefa pretendida. Por exemplo, observou-se que a utilização de um método que incluísse a função `fgetc()` para procura de palavras-chave em ficheiros de texto proporcionava desempenhos superiores quando comparada com a utilização de `fgets()`, sendo que no primeiro caso um ficheiro de 5 MB demorou, em média, menos de um segundo a ser percorrido, enquanto que no segundo caso este demorou entre 5 e 12 segundos, dependendo do conteúdo.

A obrigatoriedade de utilização de funções do Windows para realizar certas tarefas ditou a necessidade, em alguns casos, da implementação manual de certas estruturas de dados específicas do sistema, devido à impossibilidade do WinCC de reconhecer algumas bibliotecas dinâmicas do sistema.

As funções foram implementadas com sucesso, tendo em alguns casos sido necessárias pequenas alterações por parte do candidato ou do colega de equipa responsável pelo desenvolvimento do servidor para permitir uma melhor integração.

5.4. Conclusões

O desenvolvimento de blocos constituintes do sistema de gestão e supervisão foram uma introdução bastante enriquecedora ao desenvolvimento de sistemas do género, no geral, e à programação de C em e para sistemas operativos Windows. A falta de certas facilidades à criação de programas em C (devido à inexistência de certas funções) neste sistema operativo obrigaram o candidato a procurar percursos alternativos para atingir o objectivo proposto.

Ao analisar o tempo despendido com a programação das funções, é fácil concluir que o tempo gasto na programação em si é ínfimo, quando comparado com aquele utilizado a definir a estrutura de uma função ou a pesquisar métodos de abordar o problema. No entanto, considera-se que esse tempo é um investimento, facilitando o posterior desenvolvimento e manutenção das funções e ou programas criados.

6. POSTO DE TRABALHO OP25

6.1. Introdução

O posto de trabalho OP25 encontra-se situado entre o OP15 e o posto de controlo do pinhão, na linha de montagem. A função deste é a montagem de dois casquilhos seguida da montagem de uma válvula de pressão, ambos no corpo da bomba. Para realizar estas duas operações sequencialmente, o OP25 encontra-se dividido em três secções:

- Posto de carga: local onde os corpos das bombas são introduzidos na máquina;
- Posto de introdução de casquilhos: onde é realizada a montagem dos casquilhos no corpo da bomba;
- Posto de montagem de válvula: secção onde um operador introduz os elementos que compõe a válvula no corpo da bomba, para posterior prensagem automática de tampão, finalizando a montagem desta. É também neste posto que se faz a descarga do produto resultante.

Ao todo, são montados seis componentes no corpo da bomba de óleo: dois casquilhos, de dimensões idênticas, em locais diferentes, e a válvula, formada por quatro componentes. Os componentes da válvula são um pistão, uma mola, uma cavilha e um tampão. Estes elementos, com excepção dos casquilhos, têm diferenças dimensionais, dependendo do modelo de bomba a produzir, sendo que são iguais os componentes destinados aos modelos M9T e R9M. O pistão destes dois modelos contém também um filtro, ausente no das bombas Hxx. Na figura 6.1 pode-se observar dois casquilhos, em cima, com os componentes de uma válvula pertencente a uma bomba de modelo Hxx, em baixo (da esquerda para a direita, pistão, mola, cavilha e tampão).

O funcionamento do posto, a nível mecânico, e os dispositivos utilizados foram pré-determinados pela Atena e o cliente. Como tal, a descrição do funcionamento do OP25 é apenas um enquadramento do trabalho desenvolvido pelo candidato (programação do autómato e outros equipamentos), e com o que estes programas têm que lidar. Uma excepção a esta regra é o fluxo de informação e comunicação, que foi desenvolvido pelo candidato. Uma descrição do funcionamento do OP25 pode ser lida – fluxo de informação incluído – no anexo 2 (“Funcionamento do Posto de Trabalho”).



Figura 6.1: Componentes a montar pelo posto de OP25, para modelo Hxx.

Este posto de trabalho é relativamente complexo, contendo um elevado número de elementos móveis e dispositivos configuráveis. Devido a isso, a programação deste – incluindo autómato e certos equipamentos periféricos – foi atribuída ao candidato, para permitir uma aprendizagem variada.

A figura 6.2 apresenta um esquema do posto de trabalho OP25, incluindo o transportador que leva os corpos da bomba de óleo ao posto.

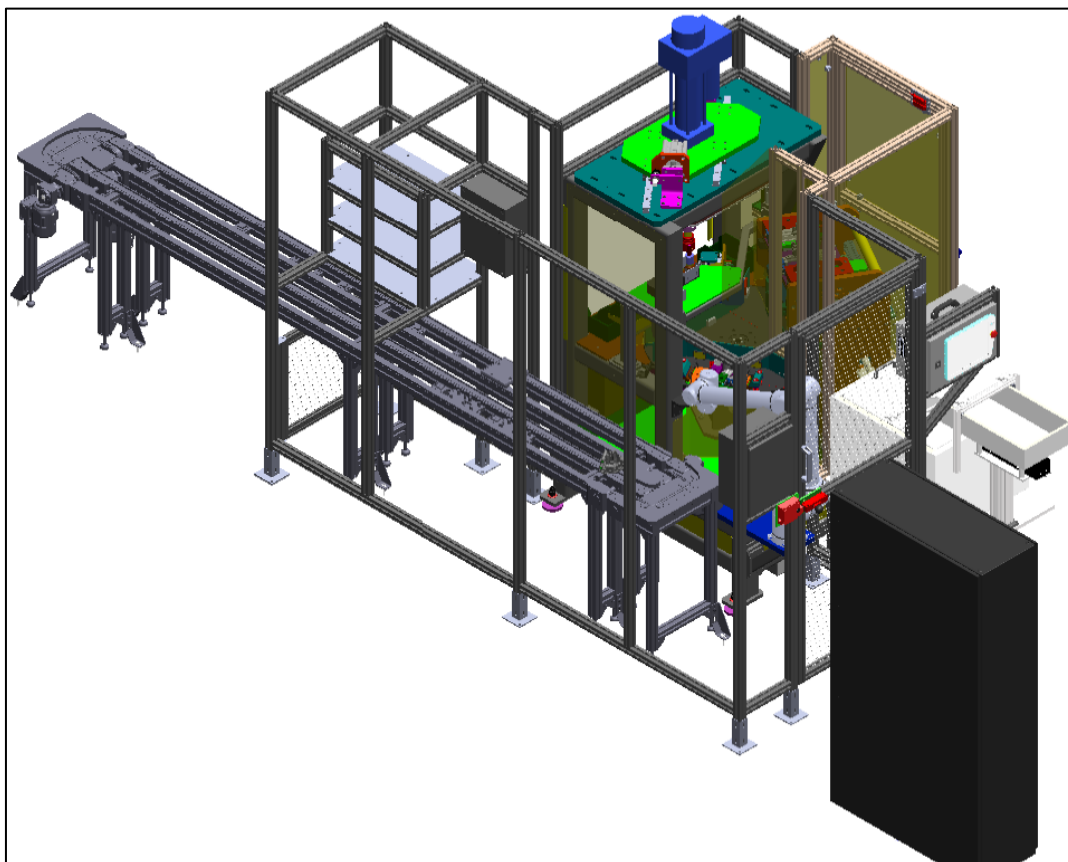


Figura 6.2: Esquema do posto de trabalho OP25, com transportador.

6.2. Programa do Autómato

O programa do autómato pode-se dividir em três grandes partes: a organização, a configuração do *hardware* e a programação propriamente dita. A boa organização e estruturação de um programa é extremamente importante, não só para facilitar o desenvolvimento, mas também para permitir uma rápida e eficiente manutenção deste, caso seja necessário.

6.2.1. Organização do Programa

A organização do programa do autómato foi feita seguindo-se as linhas gerais de funcionamento mecânico do posto de trabalho OP25 (descritas no anexo 2): uma divisão do programa em quatro secções relativas às secções da máquina (montagem dos casquilhos, montagem da válvula, posto de carga e a última para o transportador, também controlado por este PLC), com outras duas secções adicionais, uma primeira para controlos da máquina no geral, e uma segunda para comunicação com o servidor.

Cada secção contém funções (FC) e blocos de dados (DB). As primeiras consistem nas rotinas e blocos de código necessários ao funcionamento da máquina. Já os DB são locais de declaração e armazenamento de valores de memórias (variáveis). Sempre que possível associou-se um DB a uma FC, dando-lhes o mesmo nome (por exemplo, para a “FC241”, utilizou-se um “DB241”), com o objectivo de manter as variáveis contidas ao bloco de função onde estas são lidas e escritas. As vantagens dos DB incluem o agrupamento de todas as variáveis relacionadas com o funcionamento de certa parte da máquina, simplificando a procura e associação destas, e também o facto de as memórias contidas em DB serem persistentes, mesmo quando o PLC fica sem energia, ao contrário de memórias gerais (por exemplo, no caso do *bit* “M20.0”). É também possível definir valores iniciais para estas memórias, em contraste com memórias gerais, sem ser necessário primeiro escrever para estas. No programa associaram-se *user defined datatypes* (UDT) a certos DB. As UDT são estruturas, neste caso utilizadas para definir os elementos móveis no programa (cilindros, prensas e restantes) de cada secção. Deste modo, é possível associar vários DB à mesma UDT, simplificando a declaração de memórias associadas aos elementos móveis, em vez de se recorrer ao processo ineficiente de estar a declarar manualmente cada elemento em cada DB que o necessita. Pode-se observar uma relação entre UDT, DB e FC na figura 6.3 correspondente às seguranças dos elementos móveis e activação destes.

Dentro de cada secção (excepto a de comunicação com o servidor), as FC dividiram-se por tarefas:

- Geral: para acções que não se inserem nas restantes FC, relativas a cada secção de máquina;
- Segurança: que define a segurança dos elementos móveis;
- Saídas: para as saídas do autómato, nomeadamente activação de elementos móveis;
- Defeitos: para verificação da existência de defeitos com a máquina;
- Verificação de Componentes: realiza a verificação das operações realizadas pela máquina (“boas”, “más”, etc);
- Origem: para realização do ciclo de origem da máquina (descrito adiante);
- Automático: onde se encontram definidas a sequência, estados ou memórias do ciclo automático.

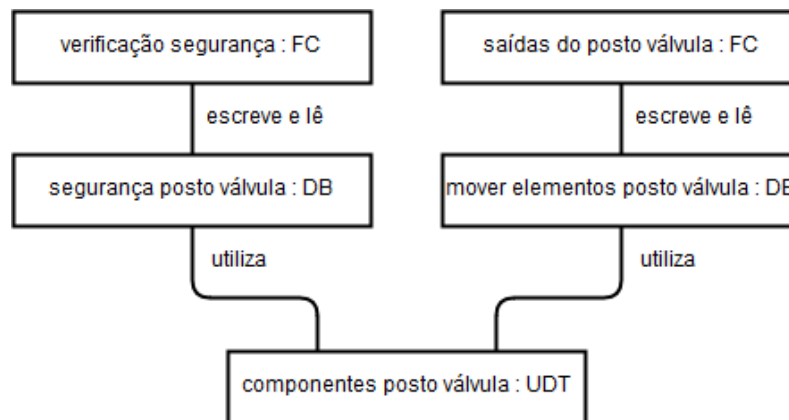


Figura 6.3: Relação entre UDT, DB e FC.

Além das FC descritas acima, existem também FC de interacção com outros subsistemas do posto de trabalho, como uma função de gestão do prato rotativo e outra para gestão do controlador de posição da prensa hidráulica, por exemplo. No caso do posto de carga existem várias funções de comunicação e comando do robô. Relativamente à secção de funções gerais da máquina, a organização exposta acima é apenas parcialmente utilizada, não existindo ciclo automático, ciclo de origem e outros, mas adicionando-se a gestão de alarmes da máquina e a comunicação com as interfaces homem-máquina. O desenvolvimento das funções de comunicação e a sua interacção com o resto do programa foi realizado por um colega de trabalho, ficando a organização desta secção a cargo deste, mas baseou-se na utilização de uma função para leitura e outra para escrita.

Utilizaram-se também, de modo limitado, FB (*blocos de funções*). As FB têm um comportamento similar às FC, mas, além de executarem operações programadas nelas, são também capazes de possuir argumento, tanto de saída como de entrada. Geralmente têm um DB associado contendo os argumentos de entrada e saída. Estas FB são utilizadas no programa em questão para monitorizar o estado dos cilindros pneumáticos e seus sensores, verificando se não

existe nenhum defeito com estes, e também para interagir com certos dispositivos (através de FB providenciadas pelo próprio fabricante desses dispositivos), como é o exemplo da prensa eléctrica (onde a FB é utilizada para todas as interações com a unidade de controlo) e o controlador da posição da prensa hidráulica (cuja função é converter a informação recebida em dados que podem ser posteriormente utilizados).

Procurou-se um equilíbrio entre a existência de poucas funções de elevada complexidade (que simplifica e limita as conexões entre funções, mas aumenta a complexidade e dificuldade de leitura de cada uma delas) e de muitas funções simples (que obrigam a muita troca de informação entre estas, aumentando a complexidade das ligações, mas simplificando o conteúdo destas). O ciclo principal do programa (OB1) é utilizado simples e unicamente para chamar as restantes funções das secções do programa, sendo que, regra geral, as funções são unicamente chamadas por ele. Numa minoria de casos, certas funções e FB chamam funções associadas, dando origem a uma organização geral mista, utilizando-se os padrões de programação de PLC “árvore” e “flor” [93].

Foram também introduzidos ciclos “OB” específicos da Siemens para não permitir que o autómato pare de executar o seu programa (comportamento por defeito), caso haja uma falha com algum do equipamento a ele ligado (falhas do dispositivo, falhas de comunicação, e afins. Estes ciclos adicionais não chamam funções, servindo apenas como “*flags*” para o PLC.

6.2.2. Configuração do Hardware

Para a configuração do hardware foi necessário definir cada equipamento no programa do autómato, definir as redes de comunicação a utilizar e associar cada equipamento à rede correspondente. Também foi necessário configurar individualmente equipamentos, para estes poderem ser localizados pelo automático e interagidos com ele.

Inicialmente definiu-se o autómato a utilizar e configurou-se o seu IP e redes (neste caso, PROFIBUS e PROFINET). De seguida pôde-se adicionar equipamentos, associando-os a cada rede. No entanto, para o autómato conseguir comunicar com estes, foi necessário adicionar ficheiros GSD – *General Station Description* – ao programa. Estes ficheiros descrevem o equipamento ao autómato, e servem como a ligação entre estes dois. Foi também necessário definir os IP de cada dispositivo, assim como o nome utilizado por estes na rede PROFINET (o que não é necessário na rede PROFIBUS). Isto obrigou à configuração dessas duas propriedades nos próprios equipamentos, de modo a que coincidam, sendo que isto pode ser feito através do programa de configuração de *hardware* ou directamente nos dispositivos. Posteriormente, foi necessário definir os endereços atribuídos a cada dispositivo (associados a memórias de entrada e saída do autómato), para este receber e enviar dados de e para os equipamentos.

A figura 6.4 apresenta um diagrama de implantação do *hardware* configurado no programa do autómato. As cartas de expansão apresentadas são ligadas directamente ao módulo de expansão e não necessitam de configuração, além da definição dos endereços relativos às memórias. O servidor, apesar de não ter sido configurado no programa do autómato, encontra-se incluído. Finalmente, em relação aos nomes dos equipamentos conectados por PROFINET, estes apresentam o nome do posto de trabalho, seguido de uma sigla identificativa da marca do dispositivo e finalmente, um valor correspondente ao primeiro endereço afecto a esse dispositivo. A título de exemplo, o equipamento identificado como “OP25SMC74” pertence ao posto de trabalho OP25, é da marca SMC e o primeiro endereço deste é o número 74.

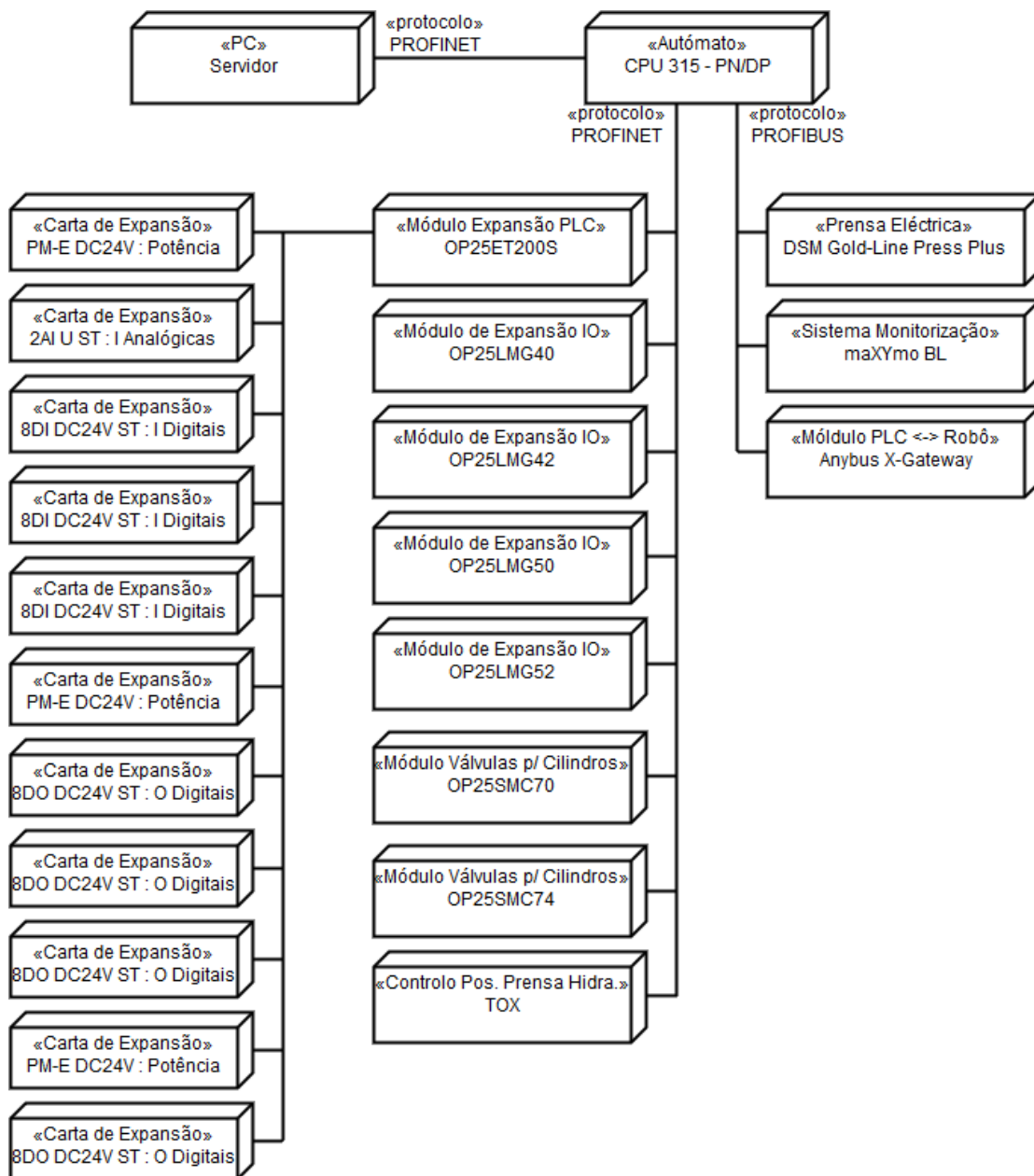


Figura 6.4: Diagrama de implantação do hardware do posto de trabalho OP25.

6.2.3. Funcionamento do Programa

Procurou-se, no programa do autómato, implementar funções e rotinas baseadas em padrões: trechos de código fáceis de reaproveitar e reimplementar no mesmo projecto ou em projectos diferentes. Por exemplo, o método de accionamento de um cilindro pneumático utilizado pode ser utilizado para todos os cilindros do posto de trabalho, assim como qualquer cilindro pneumático de qualquer posto de trabalho utilizando autómatos Siemens e, com ligeiras alterações, em todos os tipos de autómato. Ao nível mais básico, o programa faz uso liberal de realimentações para activar e desactivar memórias.

Na secção do programa de controlo geral da máquina, definiu-se principalmente o comportamento das entradas e saídas associadas às cartas de expansão, sendo que estas estão maioritariamente conectadas aos sistemas luminosos e botões e interruptores de comando do OP25. Um exemplo é a rotina de accionamento dos sinais luminosos de máquina em defeito. Se se cumprir todas as condições (máquina em serviço e alarme activado) a saída é activada e o sinal ilumina-se. É também nesta secção que se gere os modos de funcionamento da máquina (que, ao fim e ao cabo, são controlados por entradas – botões e interruptores). Este sistema tem elementos redundantes: a activação de um modo de funcionamento activa não só esse modo, de modo geral, mas também memórias de modo de funcionamento relativas a cada secção do programa. Isto é desnecessário actualmente, mas caso seja necessário introduzir condições diferentes entre secções, para estas iniciarem o seu modo manual, ou de defeito, por exemplo, torna-se mais fácil a edição do programa.

Como dito anteriormente, as secções do programa referentes às secções do posto de trabalho estão divididas em funções que realizam tarefas diferentes. Estas funções estão por vezes interligadas, sendo que a validação de uma condição numa função pode despoletar a execução de uma operação noutra função. Tem-se o exemplo da activação de um elemento móvel. É necessário garantir que existe segurança para este se mover e que foi enviado um pedido de movimentação manual, através do HMI, ou de movimentação automática, pelo ciclo automático da secção. Garantidas as condições (verificação feita na função de “saídas”), é então activada a saída para a válvula, que por sua vez fará mover o cilindro pneumático. Esta gestão está apresentada na figura 6.5. De notar que o diagrama apresentado é um caso geral e equipara-se à gestão mais complexa, incluindo interacção com o ciclo de origem e realimentação (em modo manual), para manter a válvula sempre activa. Isto faz sentido se a posição de repouso do cilindro for (neste caso) com este avançado, e se se tratar de uma válvula monoestável ou de centros aberto. No caso de uma válvula biestável, apenas é necessário enviar um pulso para esta, tornando a realimentação irrelevante para manter o cilindro em posição. No caso de se pretender recuar um cilindro, a sequência é similar, mas substitui-se as memórias de avanço pelas de recuo e vice-versa.

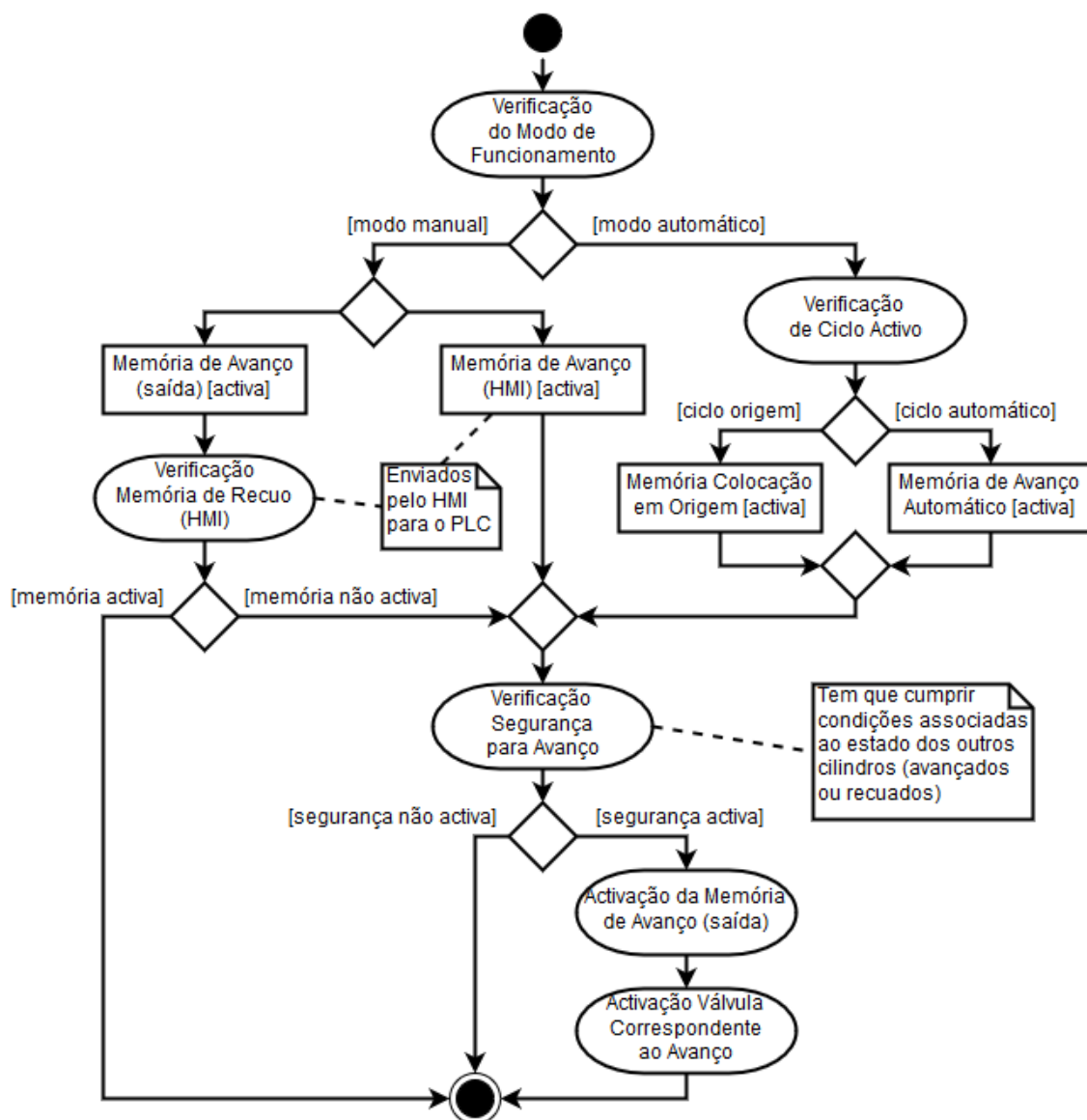


Figura 6.5: Diagrama de actividade da activação de um cilindro pneumático.

O ciclo de origem é um ciclo automático que coloca todos os elementos móveis nas suas posições de repouso. A existência desta funcionalidade é essencial, dado que se, por algum motivo, for necessário de colocar a máquina no estado inicial, não é necessário estar a movimentar manualmente cada elemento individualmente, o que seria moroso e poderia levar a que, por esquecimento, não se colocasse em repouso – por exemplo – um ou mais cilindros.

Os ciclos automáticos de cada secção foram desenvolvidos utilizando o conceito de estados e memórias, em contraste com o método passo a passo. O método passo a passo é mais fácil de implementar e muito mais fácil de fazer *debugging*, mas tem limitações a nível de versatilidade e em caso de falhas na máquina pois, se por algum motivo, esta parar a meio de um ciclo, é necessário reiniciar o sistema inteiro, já que as condições de continuar o ciclo se perdem. Já o

método de estados utiliza isso mesmo, estados dos elementos móveis e memórias armazenadas para evitar problemas em paragens, e tornar o sistema mais versátil: é possível, muito facilmente, e dentro do mesmo ciclo automático, implementar várias operações paralelas, enquanto que num sistema passo a passo é necessário criar uma rotina paralela à principal e chamada por esta num passo específico. Além dos estados existem memórias, que advêm dos primeiros, e são utilizadas para sinalizar certas ocorrências na máquina. Por exemplo, foi implementada uma memória no ciclo automático da introdução de casquilhos de modo a que esta esteja sempre activa caso a pinça de “*pick and place*” de casquilhos contenha um componente. Os ciclos automáticos das três secções estão associados entre si através do estado da mesa rotativa: apenas é possível iniciar novos ciclos se os anteriores estiverem todos concluídos. O operador, caso os ciclos estejam concluídos, pode validar o início de novo ciclo através de um botão. Isto faz rodar a mesa (e realiza a transferência de informação entre postos). Assim que a mesa esteja em posição, cada ciclo de cada secção é iniciado. Cada estado utiliza (novamente) o conceito de realimentação para controlo dos elementos móveis, quando necessário (quando as válvulas são monoestáveis). Um exemplo genérico, em linguagem Ladder, pode ser visto na figura 6.6. O endereço de entrada (I10.1) e as memórias M10.0, M10.1 e M10.2 são as condições necessárias à activação do estado (representado por M40.0). Este estado fica então sempre activo (realimentado) até que M10.1 seja desactivado ou M10.2 seja activado. Os ciclos automáticos seguem as sequências descritas nas descrições dos postos de trabalho.

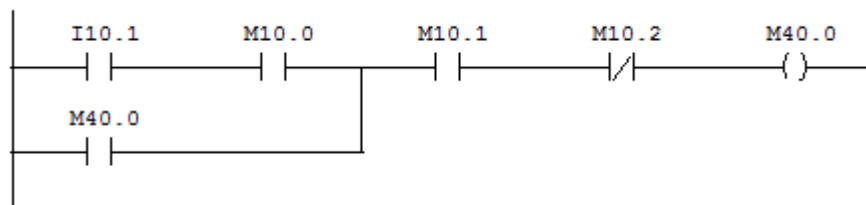


Figura 6.6: Exemplo de realimentação em Ladder.

A comunicação e gestão dos dispositivos periféricos – controlador de posicionamento da prensa hidráulica, sistema de análise de sinais desta mesma prensa, prensa eléctrica – são feitos lendo os dados que estes enviam, em bloco, processando esses dados e enviando o resultado do processamento, escrevendo nos endereços de saída, também em bloco. A gestão da prensa eléctrica é feita automaticamente por uma FB dedicada, sendo apenas necessário enviar o programa desta que se pretende executar e activando a memória de início de prensagem. A gestão do analisador de sinais limita-se a enviar pedidos de leitura, aos quais o sistema responde com os dados que está a ler dos sensores. Já a gestão do controlador pode ser feita utilizando dois métodos. O primeiro consiste em configurar programas no próprio controlador com as posições pretendidas definidas, com o autómato a enviar simplesmente o comando de execução do programa, obtendo uma resposta de programa concluído, quando este estiver finalizado. O

segundo método, que foi o utilizado, consiste em preparar apenas o controlador para receber comandos enviados pelo autómato, configurando tanto as posições pretendidas como a velocidade de movimento da plataforma da prensa internamente no PLC, recebendo do controlador a informação dessas duas grandezas (valores no momento). Isto permite uma maior flexibilidade, permitindo alterar os valores das posições e velocidade sem se necessitar de reconfigurar o controlador. É também ideal para verificação da posição e velocidade a cada instante da plataforma, o que seria impossível com o primeiro método.

O controlo do braço robótico, pelo autómato, é feito através da leitura e envio de comandos, utilizando a mesma estratégia utilizada nos restantes dispositivos periféricos. Neste caso o autómato envia comandos para pegar numa bomba de um certo modelo e colocá-la no posto de carga, recebendo do robô informações relativas ao estado deste. Estes comandos baseiam-se em programas a executar e condições de execução para o robô. O autómato envia inicialmente o número do programa pretendido, sinalizando o robô para a execução deste. As condições são utilizadas para seccionar a execução do programa: enviando todas as condições resulta na execução contínua deste, enquanto que o envio separado de condições obriga o robô a executar o programa até certo ponto e a esperar pela seguinte condição para continuar a execução do programa.

Dado que este posto de trabalho tem a capacidade de conter três bombas de óleo em diferentes estados de montagem no seu interior, foi necessário criar uma gestão das suas informações de produção. O posto tem que armazenar não só a receita de produção para cada bomba diferente, mas também os dados de produção originados tanto pela secção de introdução de casquilhos como a de montagem de válvula. Inicialmente, a informação de produção é lida ainda no transportador, mais precisamente na paleta, por um leitor RFID. Essa informação é armazenada, ao mesmo tempo que se comunica com o servidor, para este autorizar (ou não) a realização de operações de montagem nesta bomba (em caso negativo, a bomba será simplesmente transportada entre secções até ao posto de descarga). Cada secção do OP25 (posto de carga, posto de montagem de casquilhos, posto de montagem de válvula) tem armazenada a informação da bomba nelas posicionada. Findas todas as operações, a mesa rotativa será activa, de modo a transferir as bombas de uma secção para a seguinte. Neste momento, a informação associada a cada secção também sofre uma rotação, de modo a acompanhar a bomba a que pertence. Esta gestão tem que ser feita de modo cuidado, para não haver reescrita de informação ainda não transferida por informação mais recente, perdendo-se assim dados de bombas. No final, toda a informação relativa à bomba a ser descarregada do posto de trabalho é enviada para o servidor. A sequência de transmissão de informação é apresentada na seguinte sequência:

1. Leitura da informação contida na *tag* RFID;
2. Armazenamento da informação da *tag*;
3. Comunicação com o servidor para se obter, ou não, autorização de trabalho;
4. Escrita da informação da bomba no posto de carga;

5. Envio da informação referente à bomba a ser descarregada no servidor, caso esta exista;
6. Início de rotação da mesa;
7. Escrita dos dados da secção de montagem de casquilhos na secção de montagem da válvula;
8. Escrita dos dados da secção de carga na de introdução de casquilhos;
9. Limpeza dos dados na secção de carga, para aguardar nova informação vinda do transportador.

6.3. Programa do Manipulador Robótico

O programa do robô (UR10) baseia-se num ciclo infinito. Dentro deste ciclo, o manipulador fica a aguardar a informação de programa a executar a receber do autómato. Como já foi dito, a execução do programa está repartida, necessitando de condições para avançar. A informação transmitida pelo robô ao autómato inclui o da garra, aberta ou fechada, e onde está posicionada, se pegou num corpo de bomba de óleo ou não, se está em posição de repouso. A posição do manipulador é conhecida, relativamente à sua base, sendo outras posições configuradas relativamente a esta. Neste caso, configuraram-se duas zonas (mais propriamente, volumes): uma correspondente ao local de apanha do corpo da bomba de óleo no transportador e um segundo relativo ao local de posicionamento do corpo no posto de carga. Isto permite enviar ao autómato a informação da posição do robô, caso este esteja com a garra dentro destas zonas.

Este programa foi desenvolvido integralmente por um colega de trabalho, tendo-se trabalhado em conjunto para configurar a comunicação entre os programas do manipulador robótico e autómato, mais precisamente os sinais a enviar e receber por cada um.

6.4. Parametrizações de Outros Dispositivos

Além do autómato, existem outros componentes do posto de trabalho programáveis, ou configuráveis. Estes incluem o controlador do motor de posicionamento da prensa hidráulica, a prensa eléctrica e o sistema de análise de sinais da prensa de introdução de casquilhos. Estes dispositivos ora estão conectados ao PLC por PROFINET, ou por PROFIBUS.

6.4.1. Controlador de Posicionamento da Prensa Hidráulica

O motor de posicionamento da prensa hidráulica de prensagem de casquilhos faz movimentar um parafuso sem fim. A plataforma onde está a prensa está conectada a este sistema e, dependendo da tensão e correntes aplicadas ao motor, é movimentada nos dois sentidos, ao longo do fuso, a velocidades variáveis. Para controlar este motor, foi implementado um controlador (*drive*) de motores, apresentado na figura 6.7.



Figura 6.7: Controlador de motor à direita, com a fonte de alimentação do motor à esquerda.

Este controlador é comandado directamente pelo autómato, como dito anteriormente. Foi, no entanto, necessário parametrizá-lo, para que este reconhecesse os comandos enviados pelo PLC e interagisse correctamente com o motor, e para configurar o sistema de controlo eléctrico deste. Todas estas configurações foram realizadas por colegas de trabalho, partindo do pressuposto que a comunicação com o dispositivo já estava configurada, do lado do autómato.

6.4.2. Prensa Eléctrica

A prensa eléctrica utilizada para a prensagem do tampão da válvula no corpo da bomba de óleo é constituída pela unidade de actuação (a prensa eléctrica em si) e unidade de controlo. O autómato interage com esta unidade de controlo, que por sua vez envia informações à prensa relativamente à distância a avançar e força a aplicar, e analisa a informação recebida, relativamente a cotas e forças. O programa do autómato limita-se a enviar comandos de execução de rotinas alojadas na unidade de controlo, recebendo desta informações acerca do estado de execução da rotina (se está em execução ou concluída), estado da prensa (se está pronta a receber comandos, se está na posição inicial e outros) e dados acerca do resultado das operações efectuadas (se a prensagem foi “boa” ou “má”).

As rotinas alojadas na unidade de controlo são simples programas sequenciais de passos que a prensa tem que executar. Sendo que existem três modelos de bombas, foram desenvolvidas três rotinas, com cada uma delas realizando uma aproximação inicial da prensa ao corpo da bomba, seguida da prensagem propriamente dita, com cota e binário pré-definidos e, acabada a prensagem, um retorno à posição de repouso. Os valores que variam, entre programas, são obviamente a cota de introdução do tampão e a força de prensagem. Para definir estes valores, é necessário programar-se um valor óptimo de prensagem ou força, com um patamar superior e inferior de tolerância, e uma cota complementar, no caso da programação da força a aplicar, e vice-versa. Deste modo, pode-se definir que se pretende que a prensa aplique uma certa força durante a prensagem, que esta não seja superior ou inferior a certos valores, e que, independentemente da força, que a prensagem não ultrapasse uma certa cota. A aproximação da ferramenta na ponta da prensa ao corpo da bomba, por exemplo, tem uma certa distância programada, com um valor de força muito baixo, para que a prensa, durante o avanço, caso choque contra algo, não cause estragos, parando imediatamente. Durante uma prensagem, caso os valores de distância e força não correspondam ao definido, esta é dada como “má”. É também possível variar a velocidade de movimento da prensa e a evolução da distância e força num gráfico X-Y representado num ecrã da unidade de controlo.

Um diagrama de actividade de uma rotina é apresentado na figura 6.8:

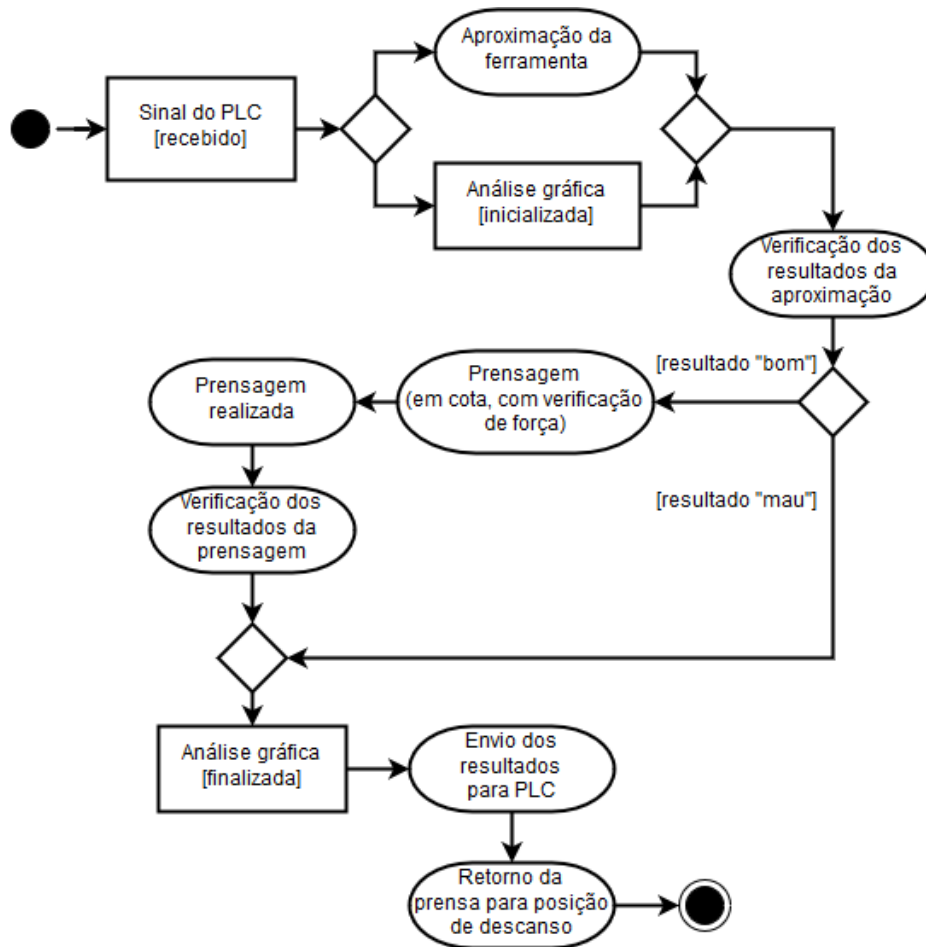


Figura 6.8: Diagrama de actividade de rotina da prensa eléctrica.

6.4.3. Sistema de Monitorização de Sinais

Dado que a prensa hidráulica não contém uma unidade de controlo e análise de dados, é necessário integrar no posto de trabalho um analisador de sinais, para se conhecer o resultado das prensagens efectuadas na secção de introdução de casquilhos. Como foi dito anteriormente, esta prensa tem nela instalados dois sensores – uma célula de carga e um potenciómetro – para analisar a cota e força de prensagem dos casquilhos no corpo da bomba de óleo. Estes sensores estão conectados a um sistema de monitorização Kistler, que analisa a evolução da cota de introdução dos casquilhos e a força que a prensa está a aplicar nestes. Estes valores são utilizados para criar um gráfico de dispersão X-Y, com a cota no eixo das abcissas e a força no das ordenadas. Os pontos são utilizados para traçar uma (figura 6.9). Parametrizando-se o sistema de monitorização, pode-se definir que, para uma prensagem ser “boa”, a curva tem que passar por uma área de valores (por exemplo, entre 2 e 4 mm de cota e 1,5 e 2 kN de força).

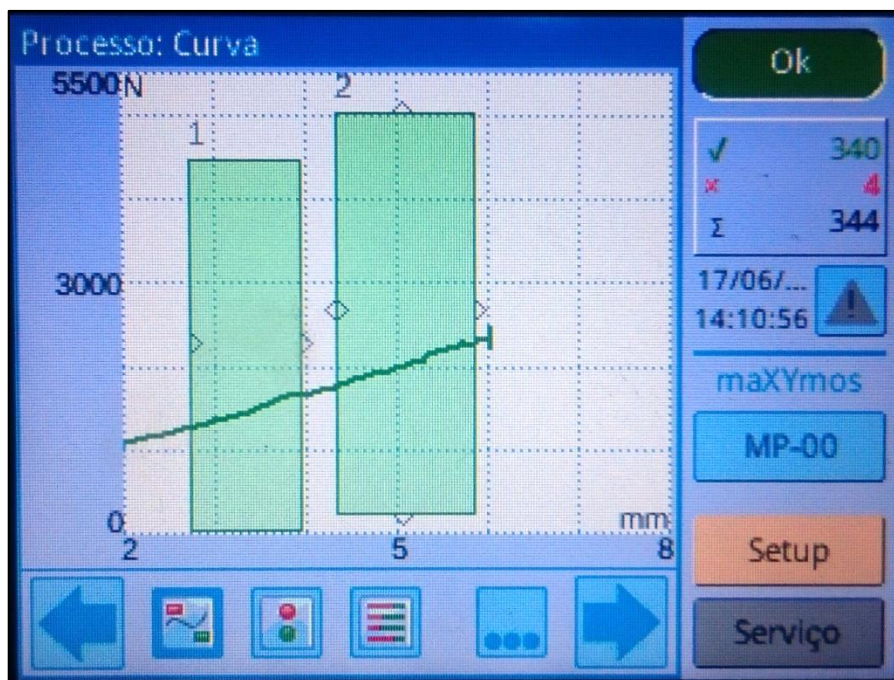


Figura 6.9: Gráfico X-Y dos resultados da prensagem de um casquilho.

Neste caso, inicialmente, existe um avanço muito grande da cota, com pouco aumento da força, correspondente à fase de aproximação da prensa aos suportes dos casquilhos na plataforma que os aloja. Depois, observa-se um crescimento praticamente linear no momento em que a prensa começa a empurrar os suportes dos casquilhos em direcção ao corpo da bomba. Finalmente, no momento de contacto dos casquilhos com a bomba, a distância percorrida pela prensa passa a ser incrementada lentamente, com a força aplicada a aumentar. A prensa recua então, com os valores da cota a regredir, sinalizando o fim da análise. Idealmente, para uma análise correcta de boa ou má prensagem, testar-se-ia a curva ao longo da aproximação, para garantir que a prensa não choca contra nada antes de prensar, seguido de novo teste na zona de crescimento rápido da força, para garantir que esta não ultrapassa ou fica aquém do esperado, assim como que a cota seja a correcta.

6.5. Interfaces Homem-Máquina

Neste posto de trabalho existem várias interfaces homem-máquina. Estas são constituídas por uma série de botões e interruptores de comando, sistemas de iluminação informativos e um ecrã táctil interactivo K7P700 Basic. Este ecrã serve para visualizar informações acerca do estado do posto de trabalho e da produção efectuada neste, assim como permitir o controlo em

modo manual dos elementos móveis, a alteração de certas memórias do programa do autómato, caso necessário, e definição de modos de funcionamento, excepto os modos manual e automático, controlado por um interruptor.

Os botões e interruptores de comando são utilizados para activar ou desactivar funções básicas do posto: colocar ou retirar a máquina de serviço, seleccionar modo de funcionamento manual ou automático, colocar a máquina em ciclo automático, realizar ciclo de origem, pedir acesso ao interior da máquina (na porta de acesso à ilha), entre outros. Existe também o botão de validação de início de ciclo, premido pelo operador a cada início de ciclo, e para validar a análise e montagem automáticas do posto de montagem de válvula.

Relativamente a luzes informativas, o posto tem um semáforo de cinco luzes, com os seguintes significados:

- Verde:
 - Intermitente: a aguardar início de ciclo;
 - Fixo: em ciclo automático.
- Vermelho:
 - Intermitente: máquina com defeito;
 - Fixo: posto de trabalho fora de serviço.
- Laranja: em funcionamento manual;
- Azul: falta de peças;
- Branco: máquina saturada.

Estes sinais luminosos servem como um aviso rápido do estado geral da máquina. Outros sinais luminosos encontram-se associados aos botões que os activam, como, por exemplo, o botão de início de ciclo, que ao ser premido ilumina-se, ou o botão de colocação da máquina em origem, que se ilumina após ser premido e enquanto o posto de trabalho está a realizar o ciclo de origem.

6.5.1. Consola Táctil

A consola táctil permite uma grande flexibilidade quanto ao envio de comandos ao autómato e leitura de informações enviadas por este, relativas ao posto de trabalho. Desenvolveu-se o programa de HMI do OP25 utilizando os programas já desenvolvidos para outros postos de trabalho da linha de montagem, de modo a manter-se a coerência entre estes, relativamente à estrutura. O programa é constituído por menus e submenus, que permitem o acesso do utilizador a vários menus de controlo do autómato (e do posto de trabalho). Como o posto está dividido em três secções, aplicou-se exactamente a mesma lógica no HMI, estando os comandos

relativos a cada posto contidos na sua própria secção da interface (excluindo o posto de carga e incluindo o transportador). Essa estrutura é apresentada na figura 6.10.

O autómato, como dito anteriormente, envia informação para a HMI, incluindo a janela actual a apresentar: caso a máquina esteja fora de serviço, o PLC envia a informação de que é necessário apresentar a janela “fora de serviço” da máquina (que é também a janela apresentada por defeito). No caso de se colocar a máquina em modo automático, será apresentada a janela correspondente, e se se colocar em modo manual, será apresentado o menu geral, para controlo directo da máquina pelo utilizador.

A janela de modo automático (figura 6.11) apresenta o estado das operações realizadas pelo posto de trabalho, as informações relativas à comunicação com o servidor (autorização de trabalho e escrita de informação neste), uma área de apresentação de alarmes e avisos, e uma caixa de texto que avisa o operador do modelo de bomba a produzir, para este saber se é necessário mudar de bloco de posicionamento no prato rotativo, no fim do ciclo automático. Procurou-se criar uma janela simples e intuitiva, para maximizar a eficiência do operador ao analisar o ecrã.

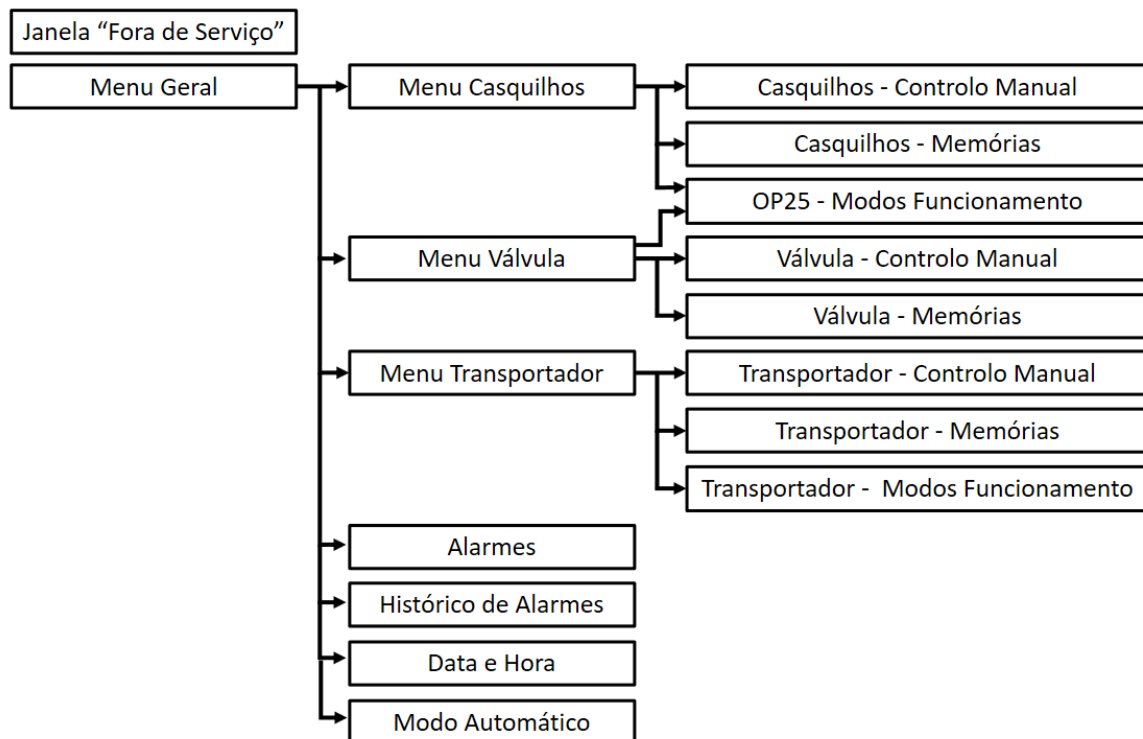


Figura 6.10: Esquema de menus e janelas do HMI do OP25.

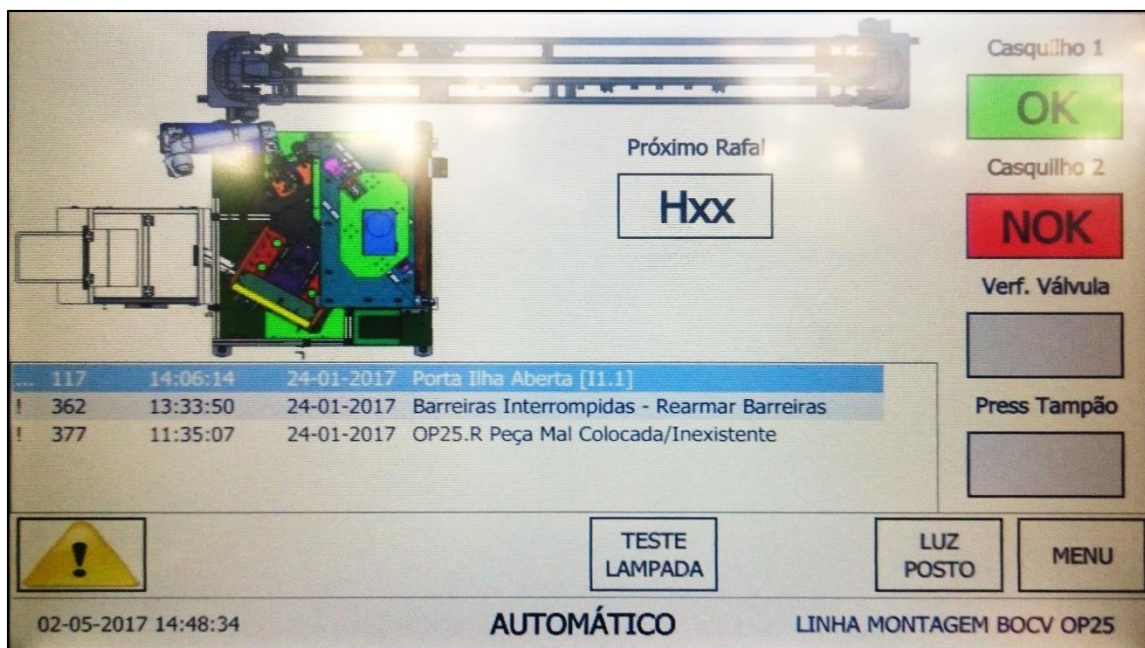


Figura 6.11: Janela de modo automático da HMI do OP25.

No modo manual, é possível controlar individualmente cada elemento móvel do posto de trabalho, exceptuando o manipulador robótico: cada menu de funcionamento manual (figura 6.12) apresenta uma listagem de elementos móveis, com informação do seu estado (avanzado e recuado nos cilindros, posição da prensa hidráulica quanto ao controlador desta, e por aí adiante) e botões de avanço, recuo ou activação. Apenas quando existe segurança (e autorização, por parte do autómato) é que os botões de avanço e recuo se tornam visíveis. O menu de modos de funcionamento permite a activação ou desactivação de modos que não o manual e automático, como por exemplo o modo de funcionamento degradado. As páginas de memória permitem alterar directamente o estado de memórias do programa do autómato, para efeitos de *debugging*.

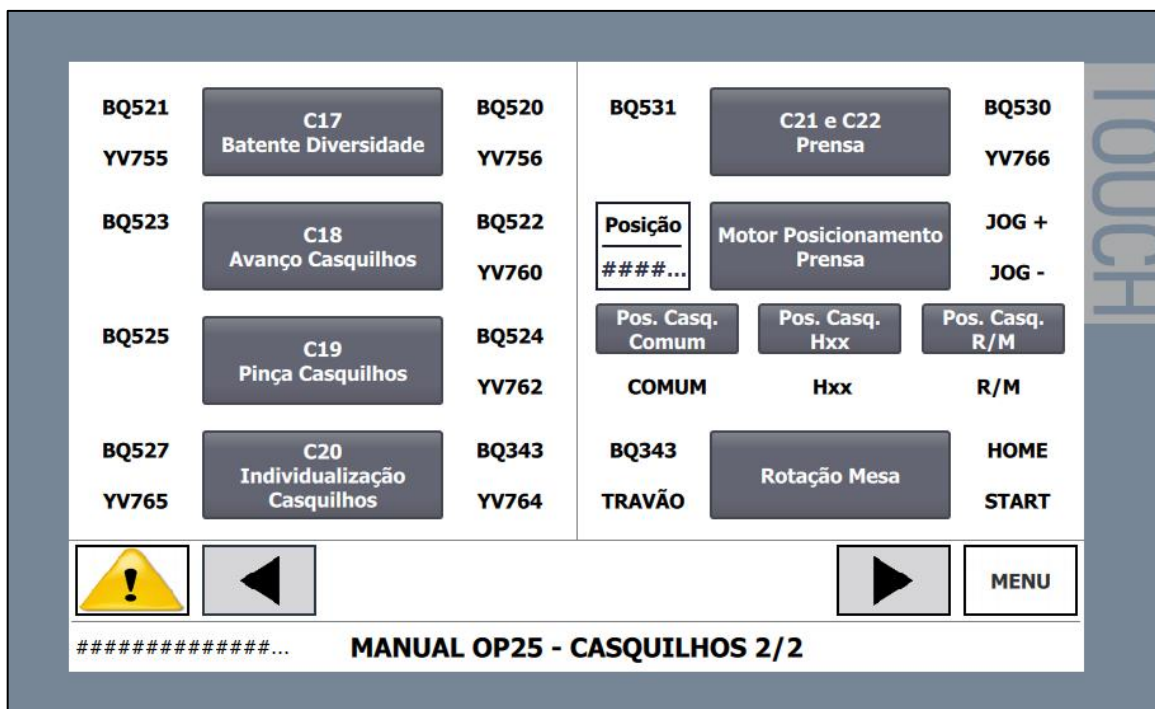


Figura 6.12: Segunda janela de controlo manual da secção de prensagem de casquilhos.

6.6. Resultados

O teste de funcionamento da máquina foi faseado. Inicialmente testou-se a comunicação entre as interfaces homem-máquina e o autómato, concluindo-se que a interacção entre ambos era a esperada. De seguida testou-se, utilizando a HMI, o accionamento manual dos elementos móveis, excluindo o posicionamento da prensa hidráulica. Revelou-se que algumas tubagens de ar comprimido foram montadas de forma errada, levando a que o suposto accionamento de um cilindro accionasse outro, ou que a tentativa de avançar um cilindro o recuasse. Estes problemas foram posteriormente resolvidos pela equipa de mecânicos. Testou-se também a montagem dos dois casquilhos e da válvula em modo manual, para se poder verificar quais os valores de avanço correctos a aplicar no caso das prensagens e quais as afinações mecânicas necessárias fazer (nomeadamente no alinhamento entre o sistema de “*pick and place*” de casquilhos, a plataforma de colocação destes e a zona de prensagem). Estes testes em modo manual foram efectuados em modelos de bomba de óleo M9T.

Os testes do funcionamento da máquina em modo automático (sem troca de informação) foram realizados parcialmente por um colega de equipa. Observou-se que existiam algumas falhas ao nível dos estados, nomeadamente comportamentos não esperados se o operador não

efectuasse a sua sequência de operações exactamente como estipulado. Relativamente ao ciclo de origem, um pequeno erro fazia com que a pinça do sistema de “*pick and place*” ficasse continuamente a abrir e fechar, não ficando o ciclo de origem concluído. Estes problemas foram atacados em conjunto com o colega de equipa mencionado anteriormente. Ao mesmo tempo que se testou os ciclos automáticos e de origem testou-se também o sistema de análise de sinais, verificando-se que, após montagem de cerca de duas dezenas de bombas, os valores de força e cota começaram a desviar-se dos valores óptimos, sem se ter realizado nenhuma alteração mecânica. Após uma observação da secção de introdução de casquilhos, concluiu-se que os suportes dos casquilhos na plataforma – que a prensa empurra para introduzir os casquilhos – tinham a peça de contacto com a prensa mal fixada, resultando no desaperto de um parafuso. Isto levava a que a única zona de contacto entre os suportes de casquilhos e a prensa fosse no parafuso, levando à sua deformação e dando origem aos resultados errados. Resolvido este problema mecânico, a prensagem dos casquilhos passou a decorrer sem problemas.

O programa do robô e a comunicação deste foram testados, com o autómato a enviar correctamente comandos ao robô, que os processou, pegou e colocou no posto de carga todos os três modelos de corpos de bombas de óleo correctamente. No entanto, devido à falta de sensores no sistema de transporte (para enviar a informação da bomba para o posto de trabalho e para este saber quando esta chegava ao local apropriado para ser manipulada pelo braço robótico) não foi possível testar o programa do autómato referente ao posto de carga, sendo que o envio de comandos baseou-se na alteração manual das memórias do autómato, obrigando a que o ciclo automático do posto de trabalho OP25 funcionasse em modo degradado, com o operador a introduzir corpos de bombas de óleo manualmente no posto de montagem de válvula. A figura 6.13 apresenta uma bomba após passar pelo posto OP25 com todas as operações deste efectuadas, mostrando o tampão da válvula prensado à esquerda (“a”) e os dois casquilhos à direita (“b” e “c”).



Figura 6.13: Resultado das operações do posto de trabalho OP25.

6.7. Conclusões

De modo geral o posto de trabalho ficou a funcionar (em modo degradado, ou seja, sem o manipulador robótico activo), tendo realizado correctamente as operações de montagem num grande número de bombas de óleo. No entanto, ficaram “por limar” muitas arestas, a maior delas todas o funcionamento do posto de trabalho com o braço robótico, assim como a fiabilidade do ciclo automático a longo prazo.

Em retrospectiva, o desenvolvimento de ciclos automáticos por estados, em detrimento de ciclos passo a passo pode à primeira vista ser uma forte aposta, devido às suas vantagens, mas, numa máquina de complexidade muito elevada, como é o caso, pode acabar por ser um problema. Muitas das dificuldades a níveis de *debugging* do ciclo automático advieram da dificuldade de seguir a evolução do ciclo no programa do autómato, já que este poderia, dependendo da situação, activar um estado num ponto do programa bastante diferente do local da operação que causou a sua activação. A utilização de um ciclo automático passo a passo simplificaria a organização do programa e tornaria a manutenção e futuras alterações, se necessário, muito mais fáceis.

Algo que não se esperava foi o facto de a parte de implementação da máquina, ou seja, a parte de teste e procura e resolução de erros fosse tão morosa. Numa máquina de grande complexidade, mesmo sabendo-se ao pormenor o funcionamento desta, é bastante difícil lidar com um problema que surja aparentemente de modo aleatório, por exemplo. A grande integração e interligação de vários componentes e dispositivos leva à necessidade de se ser metódico, percorrendo cada possível fonte de erro, testá-la, tentando reproduzir esse mesmo erro, para depois o corrigir. Este processo consome muito tempo e nem sempre se coaduna com as necessidades temporais do cliente, que procura ter a máquina a funcionar o mais rapidamente possível, já que este provavelmente tem metas próprias por imposição do mercado.

A interacção com dispositivos tão diferentes como manipuladores robóticos, controladores de motores, sistemas de análise de sinais, indexadores rotativos, entre outros, permitiu uma grande absorção de conhecimentos especializados. Estes, apesar de não poderem ser aplicados a todos os equipamentos do género, devido à gigantesca variedade de dispositivos (mesmo quando realizam funções similares) de diferentes fabricantes, permitem, pelo menos, a abertura de portas relativamente a futuras aplicações. A abordagem futura a um novo equipamento de funcionamento desconhecido será mais fácil e eficiente, facilitada pela associação dos conhecimentos aqui adquiridos.

7. OUTROS TRABALHOS DESENVOLVIDOS

7.1. Esquemas Eléctricos dos Robôs dos OP90.2 e OP90.3

Os manipuladores da linha de montagem (excluindo o do posto de trabalho OP25) têm esquemas eléctricos separados dos esquemas dos postos em que estão inseridos, já que estes possuem quadros eléctricos dedicados. Os esquemas eléctricos destes consistem na interligação dos módulos já existentes no quadro eléctrico (como por exemplo, o controlador do robô) ao manipulador robótico, assim como o accionamento de válvulas controladas por este e sensores existentes na ferramenta da ponta.

No caso do robô do posto de trabalho OP90.2 (e 90.3, já que são idênticos) é necessário apresentar o sinóptico do quadro eléctrico (que representa a localização dos vários equipamentos dentro deste), a cablagem que conecta esses módulos e o robô e as conexões de entrada e saída do robô de controlo (para válvulas e sensores) relativas à ferramenta na ponta deste, uma aparafusadora. No caso das entradas e saídas, foi necessário identificar o conector no robô afecto a estas, representar a cablagem desse conector até ao quadro eléctrico, representar a conexão entre essa cablagem e bornes com terminais, e finalmente a conexão entre esses terminais e o módulo de I/O que realiza a ligação ao controlador do manipulador. Na figura 7.1 apresenta-se a ligação entre o conector do robô de entradas e saídas, ligado a um módulo de expansão Lumberg, que por sua vez tem conectados a este “I”s” que permitem ligar dois dispositivos à mesma entrada do módulo, e três sensores e uma válvula conectados a estes “I”, devidamente identificados. A utilização destes duplicadores de saídas ou entradas foram utilizados, apesar de o módulo ter conexões suficientes para todos os dispositivos, para permitir expansão futura, caso necessário.

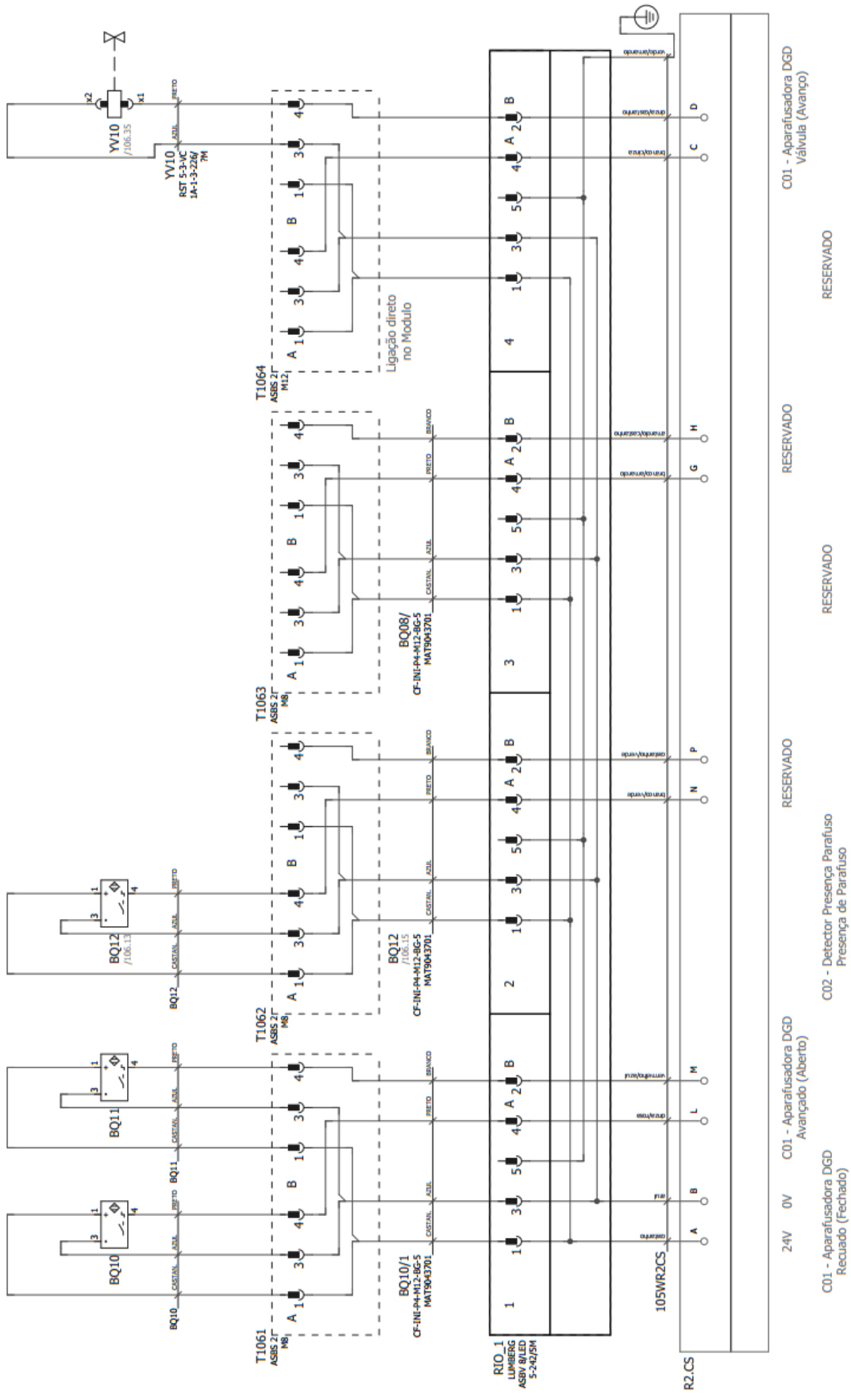


Figura 7.1. Esquema eléctrico das conexões de sensores e válvulas do robô OP90.2.

7.2. Esquema Eléctrico do Bastidor do Servidor

O servidor contém um bastidor dedicado que aloja não apenas o computador que contém o sistema, mas também periféricos (ecrã, rato e teclado) e um *switch* de rede. É também necessário alimentar este equipamento. Foi criado um esquema eléctrico do bastidor com os dispositivos e conexões entre estes necessárias. Na figura 7.2 estão apresentados a alimentação dos equipamentos, com dispositivos de corte de potência e o ventilador do bastidor. Na figura 7.3, do lado esquerdo, está representado o computador, com o *switch* do lado direito, apresentando uma das conexões Ethernet (neste caso, com o quadro eléctrico dos transportadores. A fase e neutro de alimentação são também eles providenciados por este quadro eléctrico. O bastidor em si e componentes adicionais (gaveta do servidor que aloja o rato e teclado, por exemplo) são representados pela caixa que engloba o *switch* e computador, com as referências no seu canto superior esquerdo.

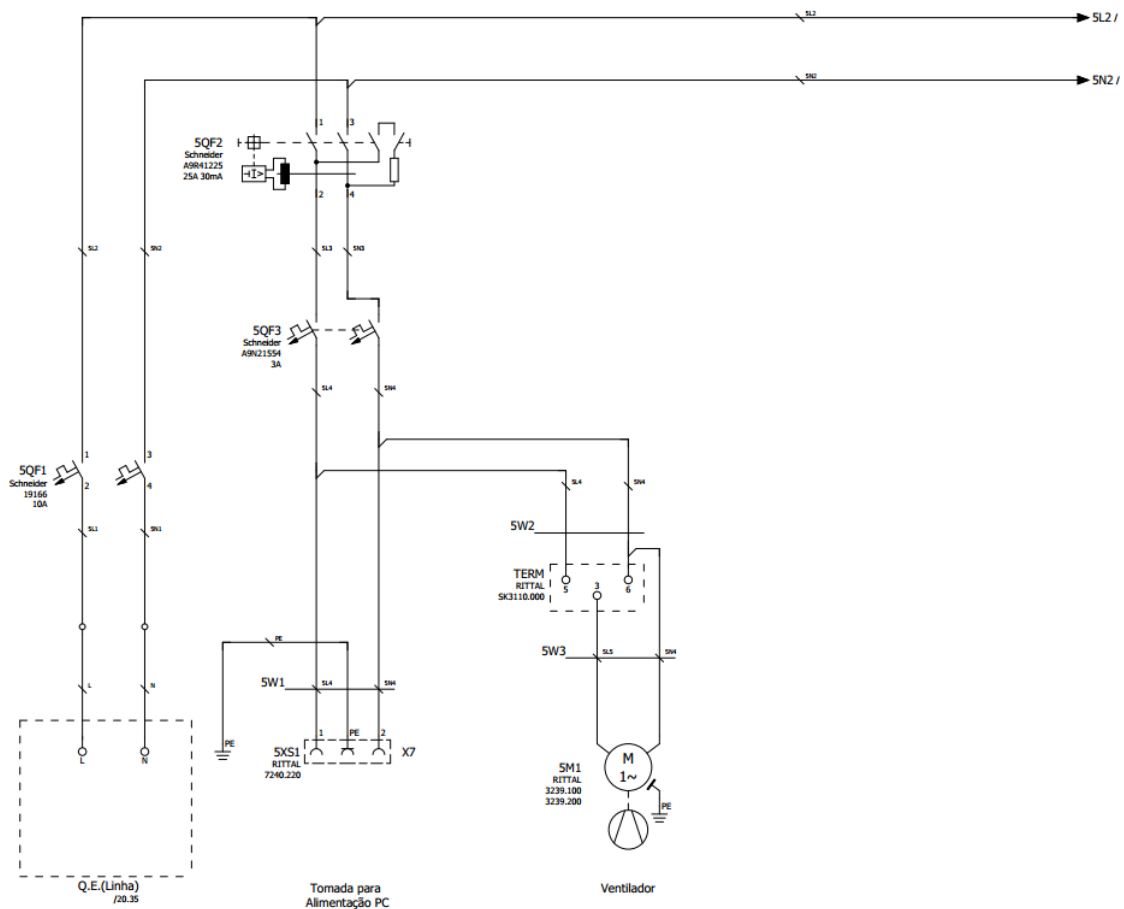


Figura 7.2: Alimentação dos dispositivos do bastidor do servidor.

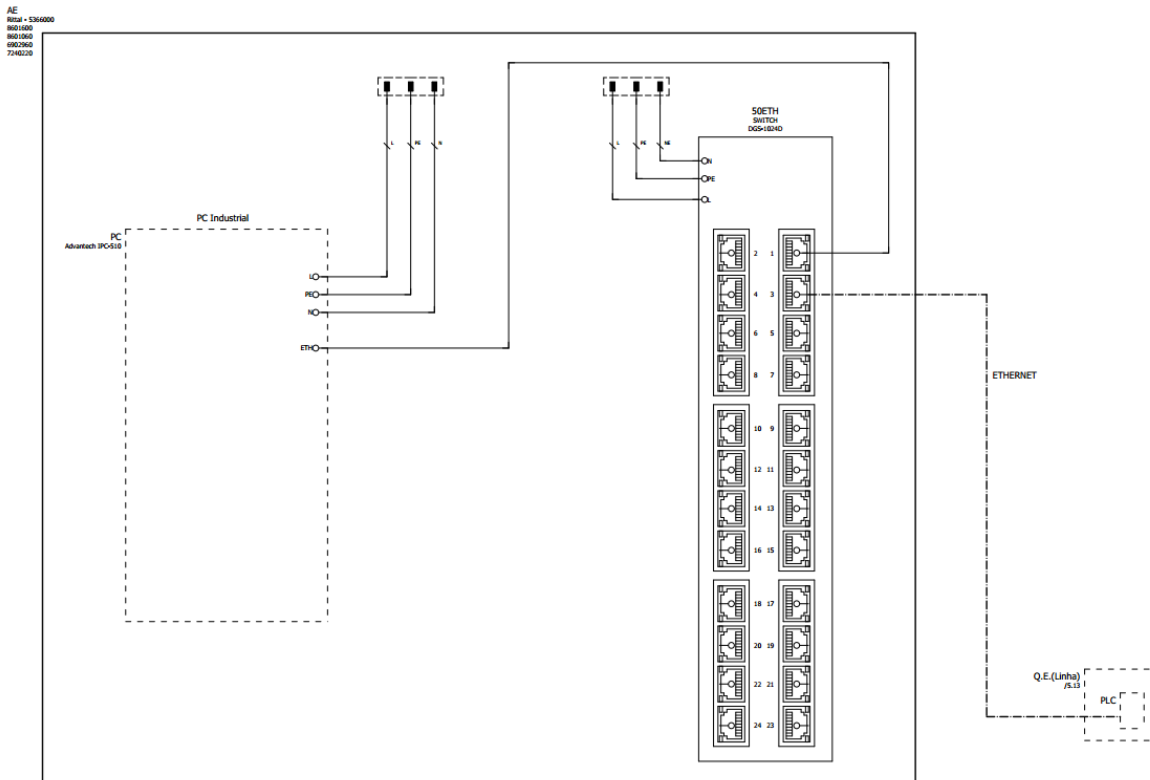


Figura 7.3: Esquema do bastidor, computador e *switch*.

7.3. HMI's das OP90 e OP40

A primeira HMI desenvolvida neste projecto (pelos colegas de equipa) foi a referente ao posto de trabalho OP70, onde se experimentou com a estrutura da interface, chegando-se à organização já apresentada na OP25, na figura 6.19. Esta estrutura foi utilizada no desenvolvimento pelo candidato da HMI do posto OP90. Neste caso, o menu divide-se em quatro secções: o posto OP90.1, o posto 90.2 e o posto 90.3, todos de aparafusamento da tampa da bomba de óleo no corpo desta, e uma última referente ao sistema de introdução de tampas no sistema automático, baseada em cinco gavetas controladas manualmente. A janela de modo automático não foi criada, pois à data de desenvolvimento da HMI ainda não havia estrutura acordada pela equipa para ela. Foi posteriormente desenvolvida por um colega de trabalho.

A interface referente ao posto de trabalho OP40, de montagem de filtro no pistão da válvula dos modelos M9T e R9M foi também ela criada de acordo com a estrutura pré-definida. No entanto, como este posto apenas tem uma secção, eliminou-se o menu geral, integrando o

menu da sua única secção nesta página (controlo manual, memórias e modos de funcionamento). A página de modo automático apresenta uma zona para alarmes e avisos, informação referente ao resultado da operação, um contador de peças boas realizadas pelo posto de trabalho e indicadores das caixas de colocação do produto final (caixa de peças boas e caixa de peças más), já que o operador tem que as esvaziar assim que estas estejam cheias. Esta janela pode ser observada na figura 7.4:

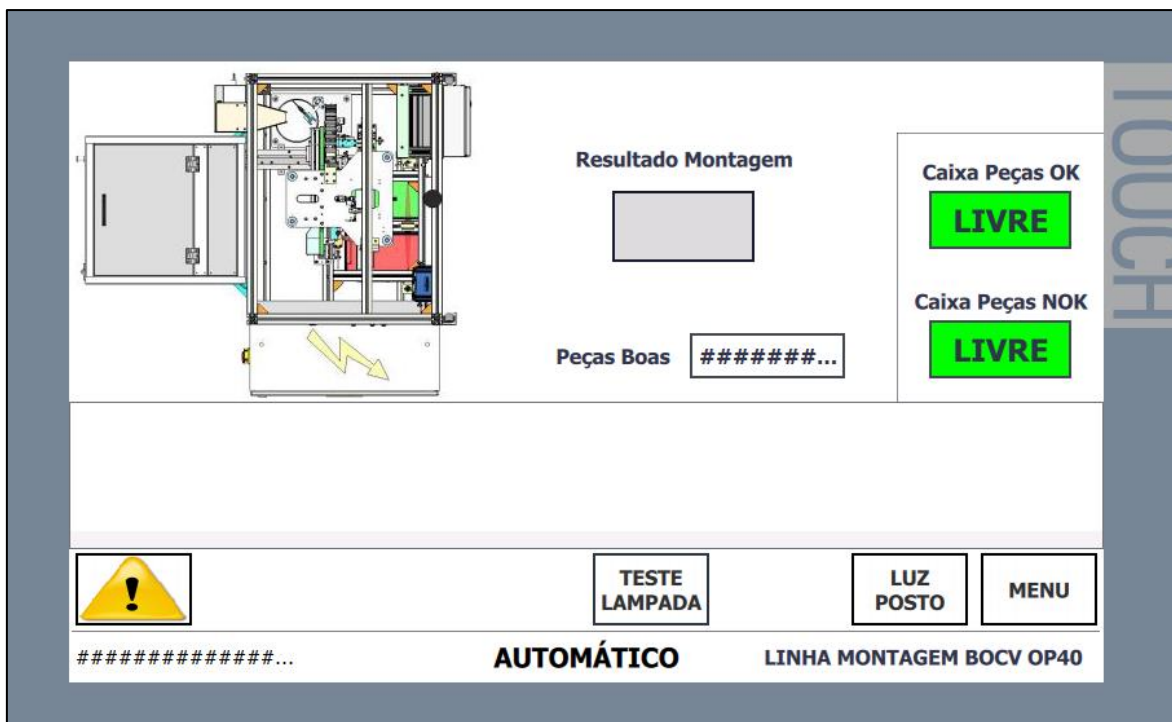


Figura 7.4: Janela de modo automático do posto OP40.

8. CONCLUSÕES

8.1. Considerações Finais

O estágio na Atena – Automação Industrial, Lda. apresentou ao candidato um leque muito alargado de tecnologias e técnicas de trabalho, além de o ter colocado ao corrente do modo de funcionamento da indústria ao nível do desenvolvimento e implementação de maquinaria.

Relativamente à utilização de um grande e variado número de *software* e *hardware* para o desenvolvimento do projecto da linha de montagem de bombas de óleo de caudal variável, foi obrigatória uma grande flexibilidade, pois surgiu frequentemente a necessidade de desenvolvimento simultâneo de programas diferentes para um mesmo sistema, utilizando ferramentas também elas diferentes. Esta variedade permitiu ainda o alargamento de horizontes quanto à programação e configuração de equipamentos, já que cada sistema e subsistema utilizou linguagens e técnicas de programação bastante dissimilares, com abordagens distintas. Considerou-se que se adquiriram bastantes conhecimentos acerca destes equipamentos, passíveis de serem utilizados no desenvolvimento de outros sistemas automáticos e dispositivos e equipamentos semelhantes.

A utilização de conceitos de programação baseados em padrões e anti-padrões nos programas do autómato do posto de trabalho OP25 a uma abordagem ao desenvolvimento do programa mais lenta, de início, comparativamente à não utilização destes, devido à maior necessidade de planificação e estruturação cuidadas. No entanto, permitiu a construção de um programas mais organizado, adaptável e expansível, caso surjam tais necessidades no futuro. Relativamente às funções desenvolvidas para o sistema de gestão de produção, foram sempre que possível desenvolvidos trechos de código possíveis de reutilizar em diversas situações, como é o caso do método de procura de informação em ficheiros utilizado em múltiplas funções. Estes trechos podem futuramente ser aplicados em novas funções e programas, mesmo que as possíveis aplicações futuras não tenham sido contempladas.

Todos os objectivos propostos no início deste projecto foram atingidos com sucesso: o desenvolvimento de sistemas específicos integrados num sistema maior (a linha de montagem), o que permitiu a aquisição de competências aplicáveis numa futura vida profissional, tanto a

nível de conhecimentos como de métodos de trabalho. O acompanhamento proporcionado pelos elementos de equipa em particular e pela empresa em geral foi uma enorme valia, não só para o trabalho desenvolvido neste estágio, mas também como uma ajuda ao crescimento profissional e pessoal do estagiário

8.2. Trabalho Futuro

O próximo passo, tendo sido concluído este estágio, é a inclusão em pleno na vida profissional, dentro da área de automação industrial. Os conhecimentos adquiridos irão, sem dúvida, ser um factor determinante e de distinção do estagiário neste mundo.

Relativamente ao projecto da linha de montagem de bombas de óleo de caudal variável em si, é necessário finalizar a implementação do sistema junto do cliente, assim como realizar a fiabilização e acompanhamento do equipamento durante o seu funcionamento normal, a fim de localizar e eliminar problemas de funcionamento, caso existam, assim como procurar meios para tornar o sistema ainda mais eficiente.

9. REFERÊNCIAS

- [1] “ENGENHARIA DE AUTOMAÇÃO INDUSTRIAL – OBJECTIVOS”. Universidade de Aveiro. Retirado de <https://www.ua.pt/course/124/?p=2>, 30 Janeiro 2017.
- [2] Central Intelligence Agency. “THE WORLD FACTBOOK – EUROPE: PORTUGAL”. Central Intelligence Agency. Retirado de <https://www.cia.gov/library/publications/the-world-factbook/geos/po.html>, 25 Janeiro 2017.
- [3] Central Intelligence Agency. “THE WORLD FACTBOOK – FIELD LISTING: GDP – COMPOSITION, BY SECTOR OF ORIGIN”. Central Intelligence Agency. Retirado de <https://www.cia.gov/library/publications/the-world-factbook/fields/2012.html>, 25 Janeiro 2017.
- [4] Sniderman, D. “BETTER FUEL EFFICIENCY THROUGH A BETTER OIL PUMP”. The American Society of Mechanical Engineers. Retirado de <https://www.asme.org/engineering-topics/articles/automotive/better-fuel-efficiency-through-a-better-oil-pump>, 27 Janeiro 2017.
- [5] Finkelzeller, K. “THE RFID HANDBOOK”, second edition. John Wiley & Sons, 2003.
- [6] Kortuem, G., Kawsar, F., Fitton, D., et al. “SMART OBJECTS AS BUILDING BLOCKS FOR THE INTERNET OF THINGS”. IEEE Internet Computing, vol. 14, n^o1, pp. 44-51, Fevereiro 2010.
- [7] Erickson, Kelvin T. “PROGRAMMABLE LOGIC CONTROLLERS: AN EMPHASIS ON DESIGN AND APPLICATION”, segunda edição. Dogwood Valley Press, LLC, 2011.
- [8] Daneels, A. & Salter, W. “WHAT IS SCADA?”. International Conference on Accelerator and Large Experimental Physics Control Systems, 1999.
- [9] Santos, A., Powell, J. “POTENTIAL OF POKA-YOKE DEVICES TO REDUCE VARIABILITY IN CONSTRUCTION”. Seventh Conference of the International Group for Lean Construction, University of California Berkeley, 26-28 Julho 1999.
- [10] Kagermann, H., Wahlster, W., Helbig, J. “RECOMMENDATIONS FOR IMPLEMENTING THE STRATEGIC INITIATIVE INDUSTRIE 4.0”. acatech – National Academy of Science and Engineering, Abril 2013.
- [11] Drath, R., Horch, A. “INDUSTRIE 4.0: HIT OR HYPE?”. IEEE Industrial Electronics Magazine, Junho 2014.

- [12] Schneider Electric. “MANUAL ELECTROTÉCNICO”. Schneider Electric España, S.A., Junho 1999.
- [13] “NORMAS INTERNACIONAIS E NACIONAIS (IEC, ISO, DIN, ...)” Pilz. Retirado de <https://www.pilz.com/pt-PT/knowhow/law-standards-norms/standards/iso-iec-standards>, 31 Janeiro 2017.
- [14] “LIST OF CENELEC NATIONAL COMMITTEES (NCs)”. CENELEC. Retirado de <https://www.cenelec.eu/dyn/www/f?p=WEB:5>, 31 Janeiro 2017.
- [15] “PORTUGAL – IEC FULL MEMBER”. International Electrotechnical Commission. Retirado de http://www.iec.ch/dyn/www/f?p=103:16:0::::FSP_ORG_ID:1025, 31 Janeiro 2017.
- [16] “IEC 60617 – GRAPHICAL SYMBOLS FOR DIAGRAMS”. International Electrotechnical Commission. Retirado de <http://std.iec.ch/iec60617>, 30 Janeiro 2017;
- [17] “IEC SYMBOL PREVIEW”. Autodesk. Retirado de http://help.autodesk.com/view/ACAD_E/2016/ENU/?guid=GUID-8C2A7E02-AC89-4164-8264-0B899573CD88, 30 Janeiro 2017
- [18] “IEC 61175-1:2015”. International Electrotechnical Commission. Retirado de <https://webstore.iec.ch/publication/22490>, 30 Janeiro 2017.
- [19] “IEC 81346-2:2009”. International Electrotechnical Commission. Retirado de <https://webstore.iec.ch/publication/7505>, 30 Janeiro 2017.
- [20] “IEC 62023:2011”. International Electrotechnical Commission. Retirado de <https://webstore.iec.ch/publication/6309>, 30 Janeiro 2017.
- [21] “IEC 62027:2011”. International Electrotechnical Commission. Retirado de <https://webstore.iec.ch/publication/6319>, 30 Janeiro 2017.
- [22] “IEC 60445:2010”. International Electrotechnical Commission. Retirado de <https://webstore.iec.ch/publication/2156>, 5 Fevereiro 2017.
- [23] “IEC/EN 60204-1”. 81346. Retirado de <http://81346.com/english/wp-content/uploads/Machinery-Directive-60204-1-and-81346-September-2010.pdf>, 6 Fevereiro 2017.
- [24] Petruzella, F. D. “PROGRAMMABLE LOGIC CONTROLLERS”, quarta edição. McGraw-Hill, 2011.
- [25] Bolton, W. “PROGRAMMABLE LOGIC CONTROLLERS”, quinta edição. Elsevier, Ltd, 2009.
- [26] Woods, D. D. “DECOMPOSING AUTOMATION: APPARENT SIMPLICITY, REAL COMPLEXITY”. Automation and Human Performance: Theory and Applications, Erlbaum, 1996, p. 3-17.

- [27] Lenze. “INTRODUCTION TO IEC1131-3 PROGRAMMING”, primeira edição. Lenze GmbH & Co KG, 2000.
- [28] Fantuzzi, C. “PROGRAMMING LANGUAGES FOR PLC: INTERNATIONAL STANDARD IEC61131-3”. Università degli Studi di Modena e Reggio Emilia. Retirado de http://www.automazione.ingre.unimore.it/pages/corsi/materialedidattico/CMA1213meccatronica/IEC61131_1.pdf e http://www.automazione.ingre.unimore.it/pages/corsi/materialedidattico/CMA1213meccatronica/IEC61131_2.pdf, 1 Fevereiro 2017.
- [29] “DIAGRAMA FUNCIONAL GRAFCET”. Instituto Superior Técnico. Retirado de <https://fenix.tecnico.ulisboa.pt/downloadFile/3779571803491/AulaT12Diagrama>, 1 Fevereiro 2017.
- [30] “FUNCTION BLOCK DIAGRAM 2” PLC Manual. retirado de http://www.plcmanual.com/images/function_block_diagram2.jpg, 5 Fevereiro 2017.
- [31] “IEC 61131 ST EXAMPLE”. Fernhill Software. Retirado de <http://www.fernhillsoftware.com/images/iec-61131-st-example.png>, 5 Fevereiro 2017.
- [32] “BILD2 9”. Beckhoff. Retirado de https://infosys.beckhoff.com/content/1033/tcpcontrol/Images/BILD2_9.gif, 5 Fevereiro 2017.
- [33] “LIST OF AUTOMATION PROTOCOLS”. Wikipedia. Retirado de https://en.wikipedia.org/wiki/List_of_automation_protocols, 1 Fevereiro 2017.
- [34] Thomesse, J. “FIELDBUS TECHNOLOGY IN INDUSTRIAL AUTOMATION”. IEEE, Institute of Electrical and Electronics Engineers, 2005, 93 (6), p. 1073-1101.
- [35] “TCP MESSAGE (SEGMENT) FORMAT”. The TCP/IP Guide. Retirado de http://www.tcpiptide.com/free/t_TCPMessageSegmentFormat-3.htm, 1 Fevereiro 2017.
- [36] Gungor, V. C. e Hancke, G. P. “INDUSTRIAL WIRELESS SENSOR NETWORKS: CHALLENGES, DESIGN PRINCIPLES, AND TECHNICAL APPROACHES”. IEEE Transactions on Industrial Electronics, vol. 56, nº 10, Outubro 2009.
- [37] Willig, A., Matheus, K., Wolisz, A. “WIRELESS TECHNOLOGY IN INDUSTRIAL NETWORKS”. Proceedings of the IEEE, vol. 93, nº 6, Junho 2005.
- [38] Han, S., Zhu, X., Mok, A. K., et al. “RELIABLE AND REAL-TIME COMMUNICATION IN INDUSTRIAL WIRELESS MESH NETWORKS”. In Real-Time and Embedded Technology and Applications Symposium (RTAS), 17th IEEE, pp. 3-12, 2011.
- [39] “ANDON (MANUFACTURING)”. Wikipedia. Retirado de [https://en.wikipedia.org/wiki/Andon_\(manufacturing\)](https://en.wikipedia.org/wiki/Andon_(manufacturing)), 1 Fevereiro 2017.

- [40] Regenbrecht, H., Baratoff, G., Wilke, W. “Augmented Reality Projects in Automotive and Aerospace Industry”. IEEE Computer Graphics and Applications, Novembro/Dezembro 2005.
- [41] Iguere, V. M., Laughter, S. A., Williams, R. D. “SECURITY ISSUES IN SCADA NETWORKS”. Elsevier, Ltd., 2006.
- [42] Malamas, E. N., Petrakis, E. G. M., Zervakis, M. “A SURVEY ON INDUSTRIAL VISION SYSTEMS, APPLICATIONS AND TOOLS”. Technical University of Crete, 2003.
- [43] Real, F. D., Berry, F. “SMART CAMERAS”. Springer Science+Business Media, LLC, 2010.
- [44] Batchelor, B. G., Whelan, P. F. “MACHINE VISION SYSTEMS FOR INDUSTRY: PROVERBS, OPINIONS AND FOLKLORE”. University of Wales, 1994.
- [45] Pan, Z., Polden, J., Larkin, N., et al. “RECENT PROGRESS ON PROGRAMMING METHODS FOR INDUSTRIAL ROBOTS”. University of Wollongong, 2012.
- [46] Pettersen, T., Pretlove, J., Skourup, C., et al. “AUGMENTED REALITY FOR PROGRAMMING INDUSTRIAL ROBOTS”. Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003.
- [47] Billard, A., Calinon, S., Dillman, R., et al. “ROBOT PROGRAMMING BY DEMONSTRATION”. Handbook of Robotics, 7 Setembro 2007.
- [48] Fryman, J., Matthias, B. “SAFETY OF INDUSTRIAL ROBOTS: FROM CONVENTIONAL TO COLLABORATIVE APPLICATIONS”. 7th German Conference on Robotics; Proceedings of ROBOTIK 2012, 21 e 22 Maio 2012.
- [49] Bolić, M., Simplot-Ryl, D., Stojmenović, I. “RFID SYSTEMS – RESEARCH TRENDS AND CHALLENGES”. John Wiley and Sons, Ltd, 2010.
- [50] Smiley, S. “ACTIVE RFID VS. PASSIVE RFID: WHAT’S THE DIFFERENCE?”. RFID Insider. Retirado de <http://blog.atlasrfidstore.com/active-rfid-vs-passive-rfid>, 5 Fevereiro 2017.
- [51] “WHICH RFID FREQUENCY IS RIGHT FOR YOUR APPLICATION?”. RFID Insider. Retirado de <http://blog.atlasrfidstore.com/which-rfid-frequency-is-right-for-your-application>, 25 Maio 2017.
- [52] Fowler, M. “UML DISTILLED: A BRIEF GUIDE TO THE STANDARD OBJECT MODELLING LANGUAGE”, 3^a edição. Adison-Wesley, Boston, 2004.
- [53] Fantuzzi, C., Bonfè, M., Secchi, C. “A DESIGN PATTERN FOR MODEL BASED SOFTWARE DEVELOPMENT FOR AUTOMATIC MACHINERY”. IFAC Proceedings Volumes, Elsevier, 2009.

- [54] Fantuzzi, C., Fanfoni, F., Secchi, C. “A DESIGN PATTERN FOR TRANSLATING UML SOFTWARE MODELS INTO IEC 61131-3 PROGRAMMING LANGUAGES”. IFAC World Congress, 2011.
- [55] Brandl, D. “DESIGN PATTERNS AND ANTI-PATTERNS”. Control Engineering. Retirado de <http://www.controleng.com/single-article/design-patterns-and-anti-patterns/571eeda95dd049436294a12b220feb38.html>, 2 Fevereiro 2017.
- [56] “PATTERNS OF LADDER LOGIC PROGRAMMING”. Contact and Coil. Retirado de <http://www.contactandcoil.com/patterns-of-ladder-logic-programming/>, 2 Fevereiro 2017.
- [57] “DATASHEET – 6ES7315-2EH14-0AB0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7315-2EH14-0AB0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [58] “DATASHEET – 6ES7151-3AA23-0AB0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7151-3AA23-0AB0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [59] “DATASHEET – 6ES7131-4BF00-0AA0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7131-4BF00-0AA0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [60] “DATASHEET – 6ES7132-4BF00-0AA0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7132-4BF00-0AA0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [61] “DATASHEET – 6ES7138-4CA01-0AA0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7138-4CA01-0AA0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [62] “DATASHEET – 6AV2123-2GB03-0AX0”. Siemens. Retirado de <https://mall.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6AV2123-2GB03-0AX0&language=en&caller=SIOS>, 3 Fevereiro 2017.
- [63] “DATASHEET – 6ES7151-8AB01-0AB0”. Siemens. Retirado de <https://support.industry.siemens.com/tedservices/DatasheetService/DatasheetService?format=pdf&mlfbs=6ES7151-8AB01-0AB0&language=en&caller=SIOS>, 3 Fevereiro 2017.

- [64] “HMI DEVICES – COMFORT PANELS”. Siemens. Retirado de https://cache.industry.siemens.com/dl/files/233/49313233/att_59649/v1/hmi_comfort_panels_operating_instructions_en-US_en-US.pdf, 20 Maio 2017.
- [65] “REXROTH INDRA DRIVE CS DRIVE SYSTEMS WITH HCS01”. Bosch Rexroth. Retirado de <http://www13.boschrexroth-us.com/goto/edc/GoTo-IndraDrive-Cs.pdf>, 20 Maio 2017.
- [66] “INDRA DRIVE C HC S02”. Bosch Rexroth. Retirado de <http://www.boschrexroth.com/dcc/Vornavigation/Vornavi.cfm?&language=en&PageID=p179794>, 5 Fevereiro 2017.
- [67] “PRODUKTKATALOG”. DSM. Retirado de http://www.grieger-automation.com/images/Automation/DSM_catalogue_2012.pdf, 5 Fevereiro 2017.
- [68] “SOPAP AUTOMATION”. SOPAP Automation. Retirado de <http://www.sopap.com/uk/home.php>, 5 Fevereiro 2017.
- [69] “AUTOFOCUS 1D AND 2D CODE READER – SR-1000 SERIES”. Keyence. Retirado de <http://www.keyence.com/products/vision/barcode/sr-1000/specs/index.jsp>, 6 Fevereiro 2017.
- [70] “PROCESS MONITORING SYSTEMS”. Kistler. Retirado de <https://www.kistler.com/?type=669&fid=108&model=download>, 20 Maio 2017.
- [71] Weiss. “PRODUCT RANGE”. Weiss GmbH, 2 Janeiro 2017.
- [72] “BOA SMART VISION SYSTEM”. Teledyne Dalsa. Retirado de http://info.teledynedalsa.com/acton/attachment/14932/f-0481/1/-/-/-/103013_BOA_dsheets.pdf, 3 Fevereiro 2017.
- [73] “UR10 TECHNICAL DETAILS”. Universal Robots. Retirado de https://www.universal-robots.com/media/1514642/101081_199901_ur10_technical_details_web_a4_art03_rls_eng.pdf, 20 Maio 2017.
- [74] “IRB 1600 – THE HIGHEST PERFORMANCE 10 KG ROBOT”. ABB. Retirado de https://library.e.abb.com/public/3b0491a94bd700a248257c71004ef393/PR10282EN_R8.pdf, 3 Fevereiro 2017
- [75] “IRB 2600 – INDUSTRIAL ROBOT”. ABB. Retirado de https://library.e.abb.com/public/b48a19048143781b482577bb00369543/ROBO142EN_B%20LR.pdf, 3 Fevereiro 2017.
- [76] “IRB 4600 – INDUSTRIAL ROBOT”. ABB. Retirado de https://library.e.abb.com/public/eadd388d8bec4b75a3342026c1631b5e/IRB%204600%20ROB0109EN_H.pdf, 3 Fevereiro 2017.

- [77] “RFID 13.56 MHZ”. IFM. Retirado de http://www.ifm.com/ifmnz/web/pmain/060_020_030.html, 3 Fevereiro 2017.
- [78] “TIA PORTAL – YOUR GATEWAY TO AUTOMATION IN THE DIGITAL ENTERPRISE”. Siemens. Retirado de <http://www.industry.siemens.com/topics/global/en/tia-portal/Pages/default.aspx>, 3 Fevereiro 2017.
- [79] “TIA PORTAL – AN OVERVIEW OF THE MOST IMPORTANT DOCUMENTS AND LINKS – CONTROLLER”. Siemens. Retirado de <https://support.industry.siemens.com/cs/document/65601780/tia-portal-an-overview-of-the-most-important-documents-and-links-controller?dti=0&lc=en-WW>, 3 Fevereiro 2017.
- [80] “STEP 7 BASIC V13 SP1”. Siemens. Retirado de http://stest1.etnetera.cz/ad/current/content/data_files/automatizacni_systemy/mikrosystemy/simatic_s71200/manually/gsg_step7-basic-v10-5_2014-12_en.pdf, 3 Fevereiro 2017.
- [81] “PROGRAMMING WITH STEP 7”. Siemens. Retirado de https://cache.industry.siemens.com/dl/files/056/18652056/att_70829/v1/S7prv54_e.pdf, 3 Fevereiro 2017.
- [82] “WINCC V7.3 – WINCC: WORKING WITH WINCC”. Siemens. Retirado de https://cache.industry.siemens.com/dl/files/925/102754925/att_62020/v1/WinCC_Working_with_WinCC_en-US_en-US.pdf, 3 Fevereiro 2017.
- [83] “STATEMENT LIST (STL) FOR S7-300 AND S7-400 PROGRAMMING”. Siemens. Retirado de <http://iat.fs.cvut.cz/109/files/S7/S7-SW2.pdf>, 3 Fevereiro 2017.
- [84] “EPLAN ELECTRIC P8 – GETTING STARTED”. Eplan Software & Service GmbH & Co. Retirado de http://149.237.200.202/fileadmin/dateien-CA/Beginner-Guide_P8_enUS_NorthAmericanStyle_NFPA.pdf, 3 Fevereiro 2017.
- [85] “INSPECT EXPRESS SOFTWARE USER’S MANUAL”. Teledyne Dalsa. Retirado de http://gb.daheng-image.com/cn/products/teledynedalsa/manual/DALSA_boa_Manual_EN.pdf, 3 Fevereiro 2017.
- [86] “TECHNICAL REFERENCE MANUAL”. ABB. Retirado de https://library.e.abb.com/public/688894b98123f87bc1257cc50044e809/Technical%20reference%20manual_RA-PID_3HAC16581-1_revJ_en.pdf, 3 Fevereiro 2017.
- [87] “OPERATING MANUAL – ROBOTSTUDIO”. ABB. Retirado de https://library.e.abb.com/public/244a8a5c10ef8875c1257b4b0052193c/3HAC032104-001_revD_en.pdf, 3 Fevereiro 2017.
- [88] “PROFIBUS – COMPREHENSIVE PROTOCOL OVERVIEW”. RTA Automation. Retirado de <http://www.rtaautomation.com/technologies/profibus/>, 3 Fevereiro 2017.

- [89] “PROFINET IO – PROFINET UNPLUGGED – AN INTRODUCTION TO PROFINET IO”. RTA Automation. Retirado de <http://www.rtaautomation.com/technologies/profinet-io/>, 3 Fevereiro 2017.
- [90] “TCP/IP PROTOCOL ARCHITECTURE”. Microsoft. Retirado de <https://technet.microsoft.com/en-us/library/cc958821.aspx>, 4 Fevereiro 2017.
- [91] Teledyne Dalsa. “BOA INSTALLATION MANUAL”. Teledyne Dalsa Incorporated, 17 Setembro 2014.
- [92] Armendariz, D. “APERTURE AND F-NUMBER”. Digital Media E-10: Exposing Digital Photography, Harvard Extension School. Retirado de <http://digitalphotography.exposed/slides/aperture.pdf>, 22 Maio 2017.
- [93] Fuschs, J., Feldmann, S., Legat, C. et al. “IDENTIFICATION OF DESIGN PATTERNS FOR IEC 61131-3 IN MACHINE AND PLANT MANUFACTURING”. Proceedings of the 19th World Congress of The International Federation of Automatic Control, Cape Town, África do Sul, 24-29 Agosto 2014.

ANEXOS

1. Organizações de Esquemas Eléctricos

Apresentam-se de seguida os vários tipos de organização de esquemas eléctricos: por referência, por componente e por função, respectivamente. Estes são apenas exemplos, não tendo sido desenvolvidos para o projecto das linhas de montagem de bombas de óleo de caudal variável.

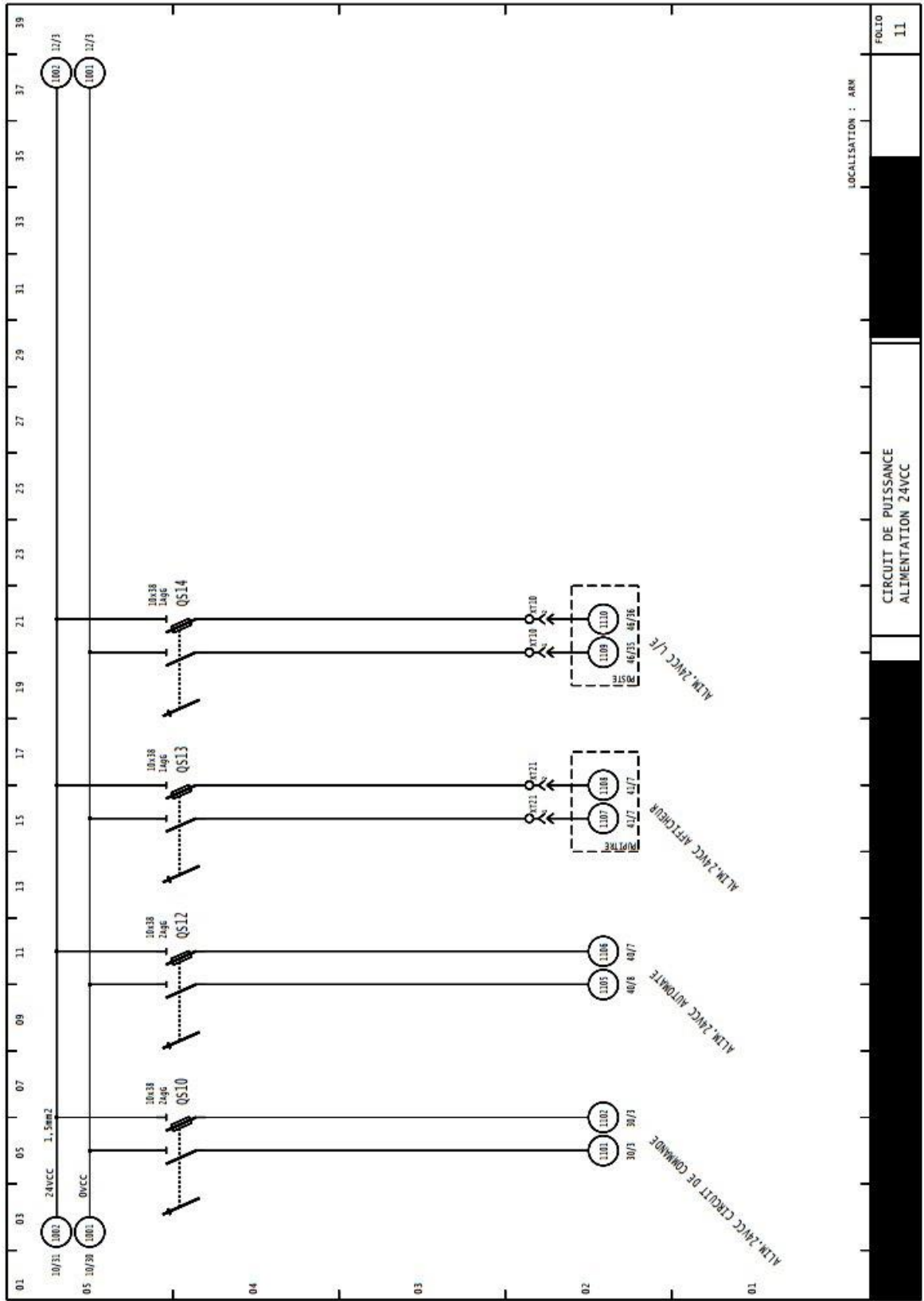
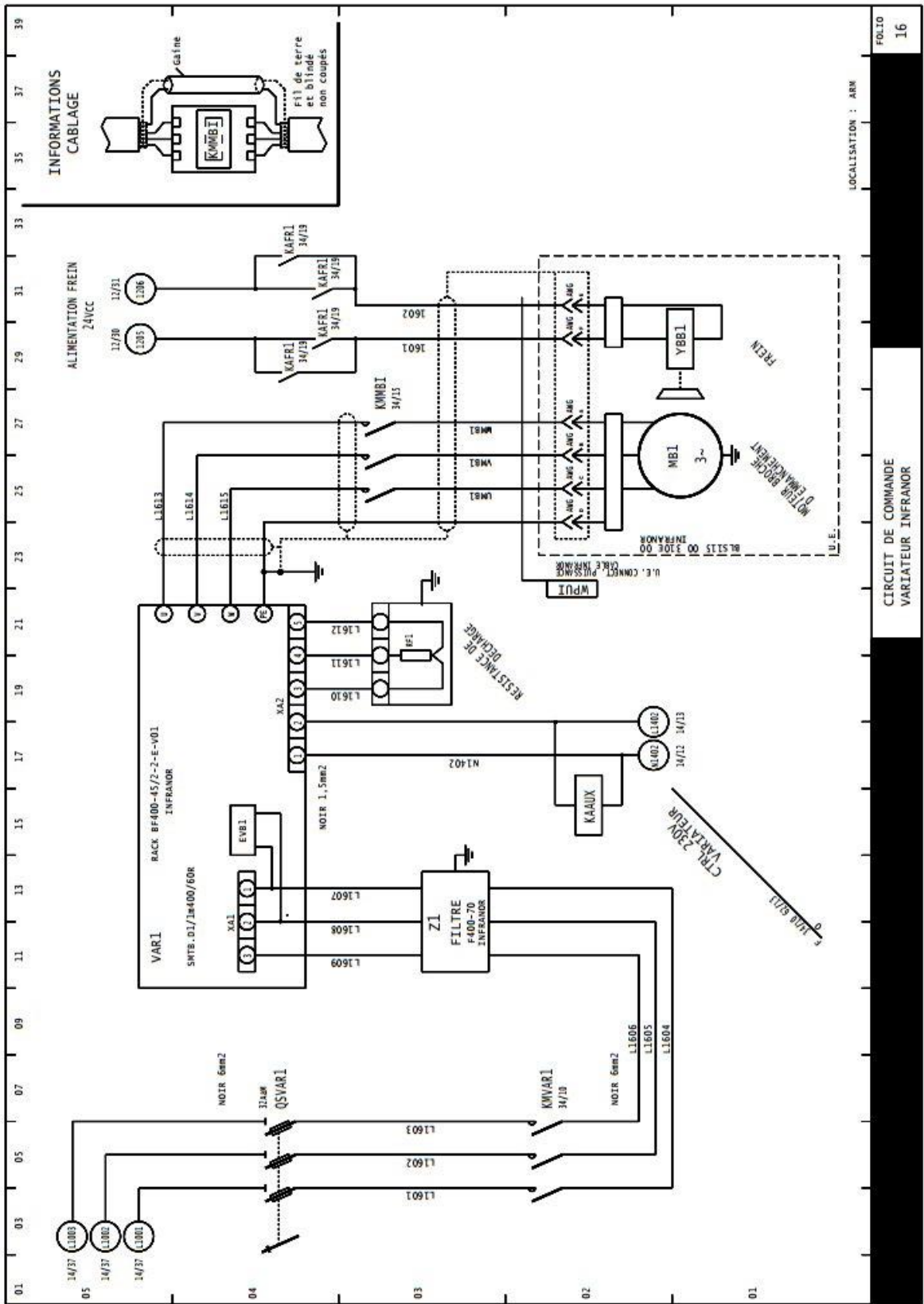


Figura 1.6: organização por componente (2).



LOCALISATION : ARN

Figura 1.8: organização por componente (4).

2. Funcionamento do Posto de Trabalho

O posto de carga é constituído por um manipulador robótico Universal Robots UR10, cuja tarefa é pegar no corpo da bomba, vindo do transportador que liga o posto de trabalho OP25 e o anterior, em paletes, e colocá-lo no local apropriado, no interior da máquina. O segundo posto é constituído por um alimentador de casquilhos vibratório, um sistema de “*pick and place*” de casquilhos que os transporta da saída do alimentador para uma plataforma transportadora, que os leva ao local de introdução na bomba de óleo, outro sistema para bloqueio da posição do corpo da bomba de óleo, de modo a que ela se mantenha estática durante a introdução dos elementos, e finalmente uma prensa hidráulica de posição variável – um controlador associado a um motor permitem variar a sua posição – que realiza a montagem dos casquilhos. Já o posto de montagem da válvula é constituído por uma prensa eléctrica, de prensagem do tampão que encerra a dita válvula no corpo da bomba e um sistema de verificação dos componentes constituintes dessa válvula de pressão, para garantir que estes foram correctamente introduzidos pelo operador, antes da prensagem. No final, esse mesmo operador retira o corpo da bomba do posto de trabalho e coloca-o no segundo transportador, dando continuidade ao processo de montagem da bomba.

Para interligar estas secções da máquina entre si existe um prato rotativo accionado por um indexador de três posições (com cada posição distanciada 120° da seguinte), que, ao ser activado, leva a bomba de uma secção para a seguinte. Cada posição no prato tem um local de encaixe da bomba constituído por um bloco amovível (figura 2.5). Cada secção do posto de trabalho é capaz de realizar operações independentemente, permitindo a montagem simultânea de duas bombas de óleo, com uma terceira a ser introduzida na máquina, obrigando a que o prato rotativo apenas rode quando todas as operações em todas as secções do OP25 se encontram concluídas. Um esquema do transporte de corpos de bombas de óleo entre secções do posto OP25 encontra-se apresentada na figura 2.1, com o fluxo de bombas representado por setas de traçado contínuo, e a traçado intermitente a rotação do prato, levando o local de encaixe (após ser retirada a bomba na descarga) do posto de prensagem da válvula para o posto de carga.

Se o posto de trabalho estiver a funcionar em modo degradado (ou seja, com o manipulador robótico desactivado) a introdução de corpos de bombas é feita no posto de montagem de válvula, pelo operador, após retirar a última bomba trabalhada desse mesmo posto, tornando o posto de carga num simples local de passagem.

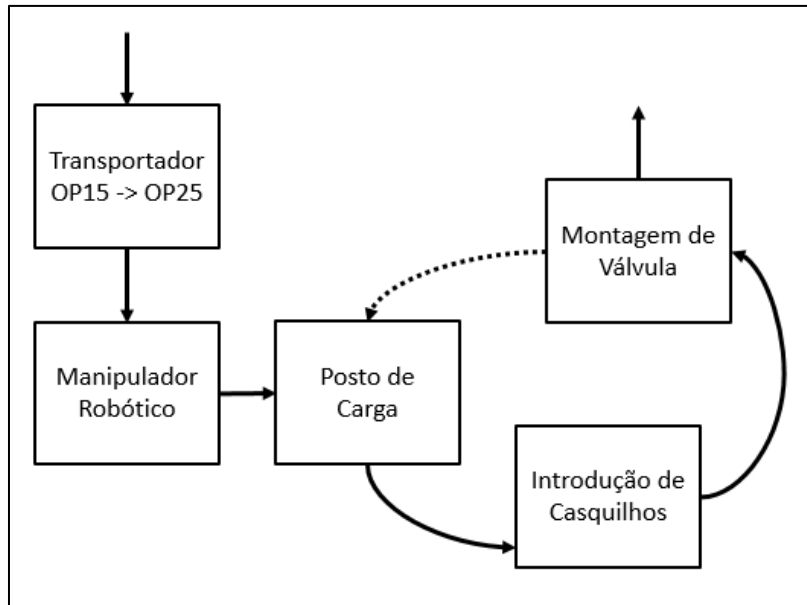


Figura 2.1: Fluxo de corpos de bombas ao longo do OP25.

Na figura 2.2 é possível visualizar-se um esquema do posto de trabalho, excluindo o robô (que estaria fixo no suporte “a” na parte inferior da imagem) e alimentador de casquilhos. A meio está representado o prato rotativo (“b”) com os locais de colocação dos corpos das bombas (“c”), sendo apenas um visível, na secção apresentada no centro da imagem, em baixo. Esta secção é o posto de carga do sistema. Em cima, à esquerda, pode-se ver o posto de montagem de válvula, com a prensa (“d”) suportada por pilares, a ferramenta de prensagem na sua ponta (objecto circular “e”) e os restantes elementos dessa zona obscurecidos. Em cima, à esquerda, contido por painéis (“f”), está contido o sistema de montagem de casquilhos. A figura 2.3 apresenta outra vista do mesmo esquema do posto de trabalho, sendo mais visível o posto de montagem da válvula, com a figura 2.4 representando o posto de montagem dos casquilhos, completo, à esquerda, e com certos elementos retirados para permitir visualizar o seu interior, à direita.

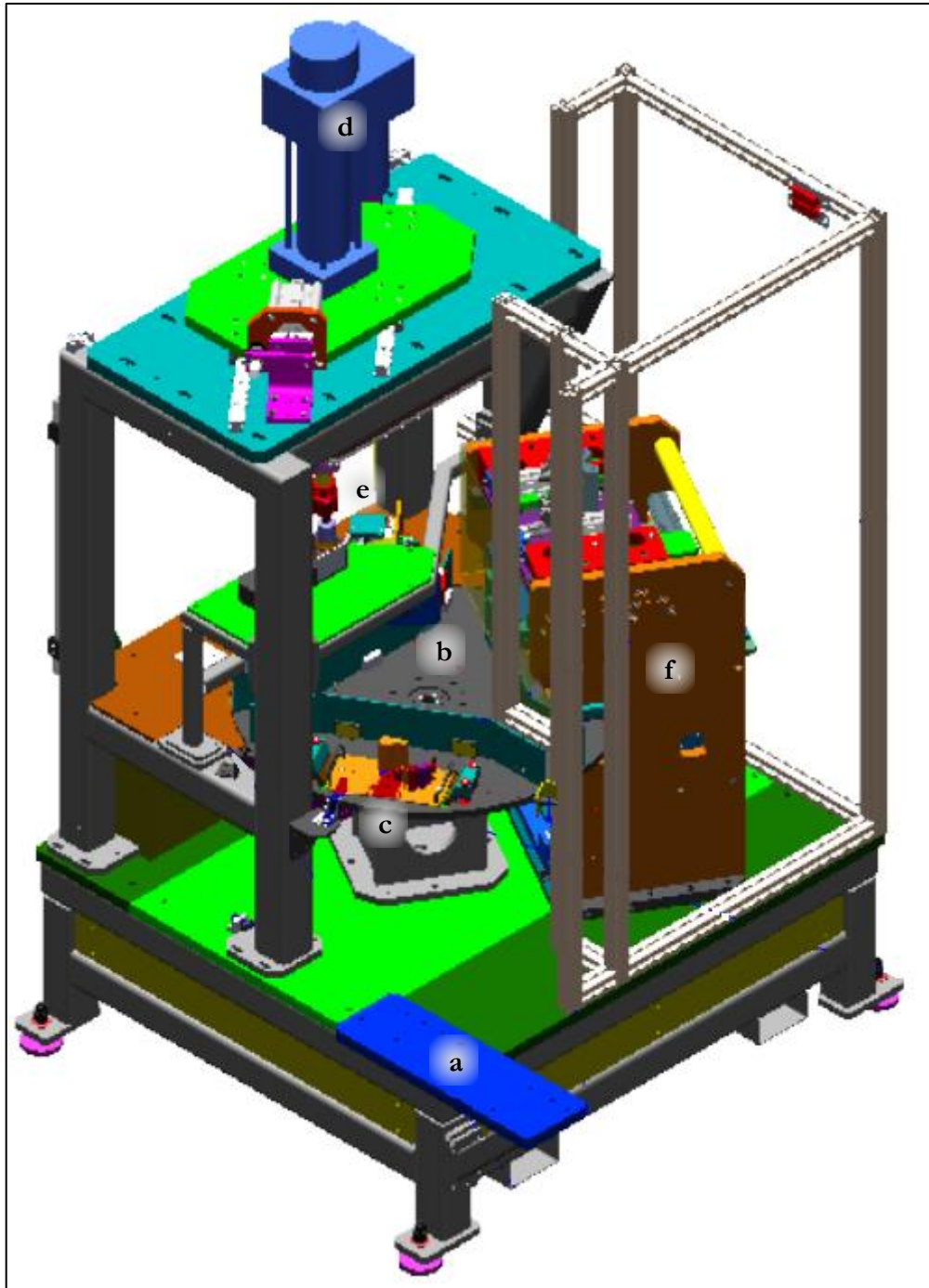


Figura 2.2: Esquema do posto de trabalho OP25, visto do posto de carga.

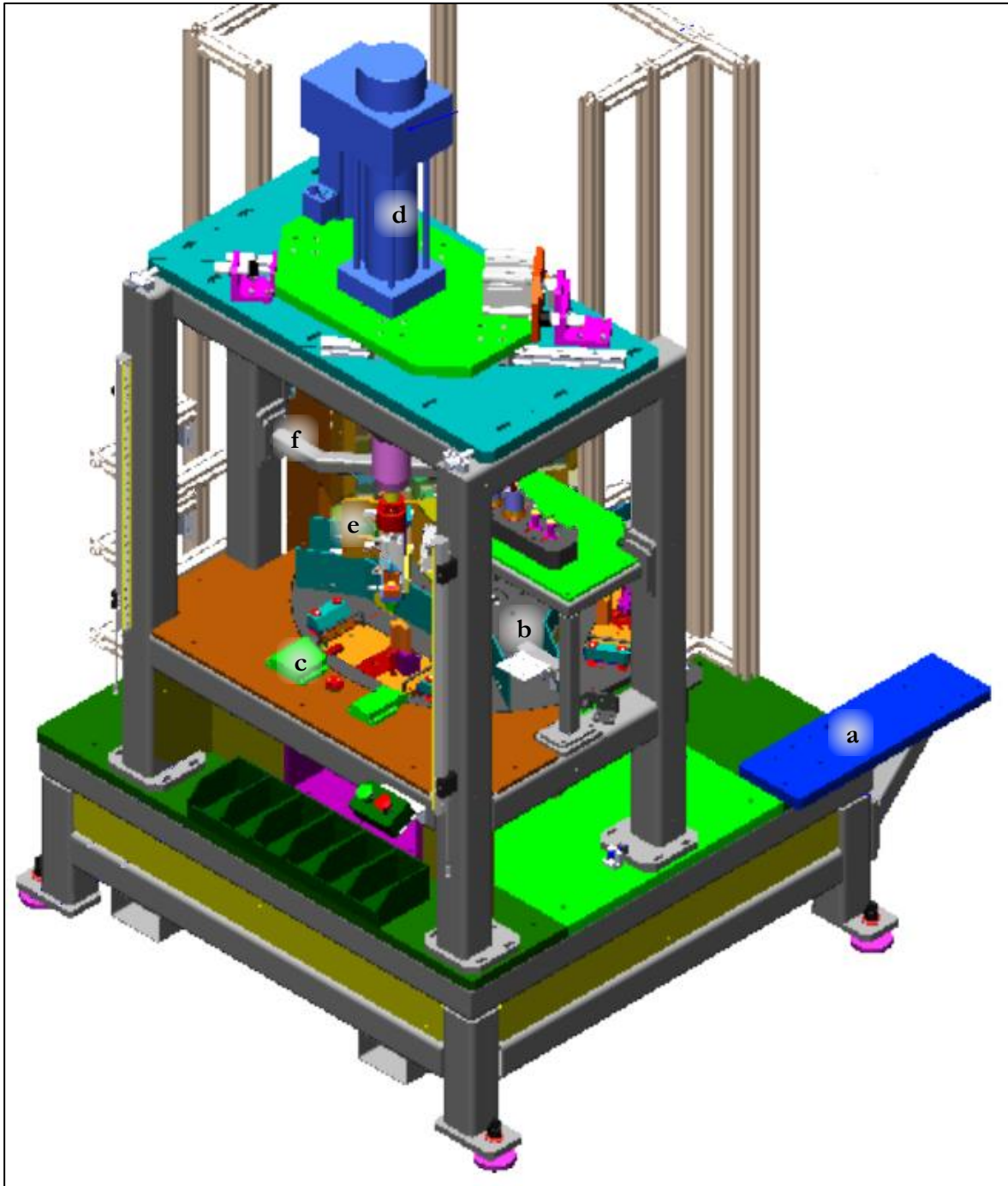


Figura 2.3: Esquema do posto OP25, visto do posto de montagem de válvula.

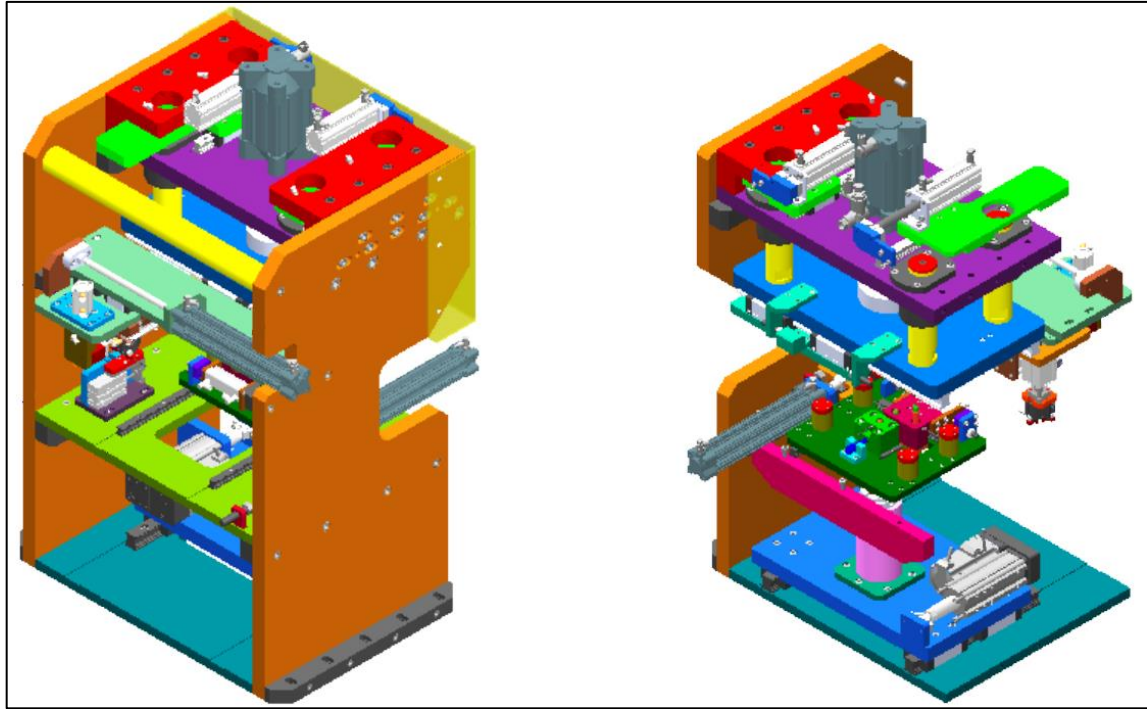


Figura 2.4: Sistema de montagem de casquilhos.

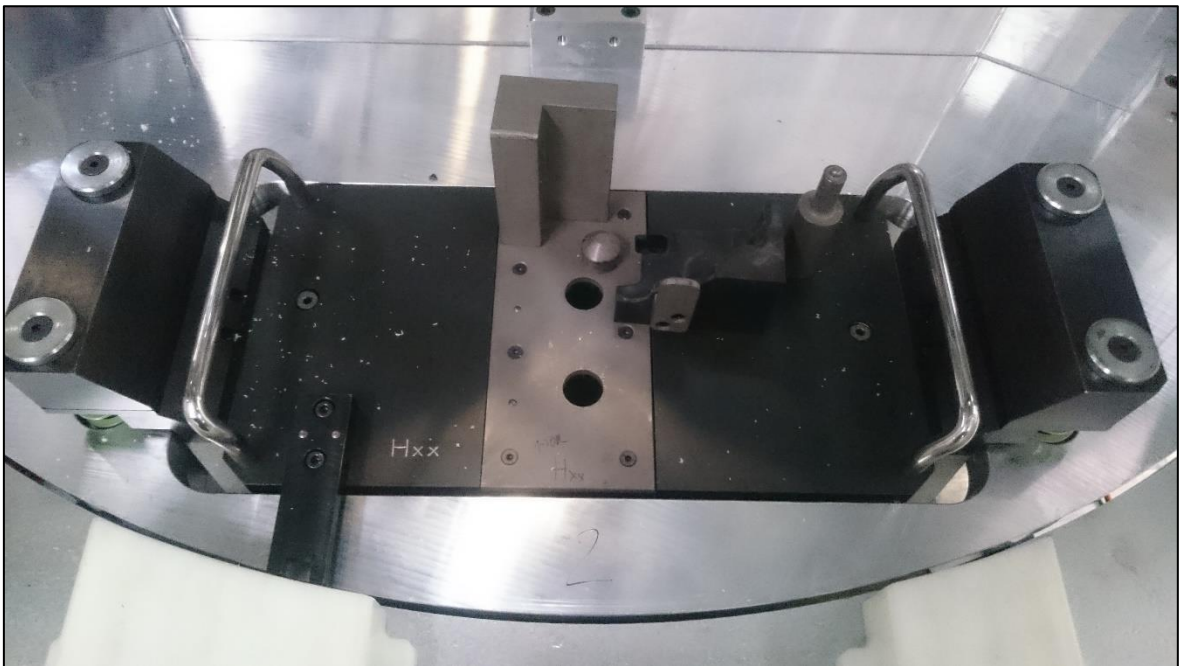


Figura 2.5: Bloco de posicionamento de corpos de bombas de óleo Hxx.

O posto de trabalho funciona utilizando um autómato, que lê o estado dos sensores, acciona válvulas (que movimentam cilindros) e envia comandos a outros dispositivos programáveis (robô, controlador de posicionamento da prensa hidráulica, prensa eléctrica). É importante mencionar que dois dos sensores presentes no posto de trabalho, um potenciómetro e uma célula de carga, colocados na prensa de montagem de casquilhos, estão conectados a um analisador de sinais, de modo a verificar se a prensagem foi bem realizada. Esta gestão na prensa eléctrica é feita internamente, por um controlador próprio a esta associado.

Tal como o resto da linha, a variedade de modelos a montar exige que o posto de trabalho consiga lidar com as diferenças geométricas e de componentes destas. Como tal, uma quantidade considerável de elementos do OP25 são reposicionados automaticamente quando existe troca de modelos a produzir, assim como mudança manual de ferramentas – a ferramenta na ponta da prensa e a localizada na ponta do sistema de verificação de componentes – no posto de montagem de válvulas. Os blocos de posicionamento das bombas no prato giratório também têm que ser substituídos manualmente, pois estes são diferentes para bombas de óleo distintas. É possível ter cada secção da máquina a trabalhar um modelo de bomba diferente, permitindo produções unitárias.

2.1. Posto de Carga

Os únicos elementos móveis nesta secção do OP25 são o manipulador robótico e componentes associados. Este tem uma garra com quatro elementos móveis controlados por cilindros pneumáticos, que por sua vez são accionados por válvulas controladas pelo robô.

Relativamente ao seu ciclo de funcionamento, o robô posiciona a garra acima do local de paragem das paletes que transportam os corpos das bombas de óleo no transportador, à espera que uma paleta chegue. Nesse momento, o robô abre a garra (avançando os cilindros) e engata-a no corpo da bomba, finalizando com o fecho da garra. O robô afasta-se então do local, levando a peça ao local de carga, encaixando-a no bloco de posicionamento, com um sensor laser a verificar se o corpo ficou bem colocado.

2.2. Posto de Introdução de Casquilhos

Esta secção é bastante complexa, contabilizando-se doze cilindros pneumáticos diferentes para movimentar elementos variados, com mais dois hidráulicos para accionar a prensa de casquilhos. O sistema de “*pick and place*” de casquilhos é formado por uma pinça (no centro, à direita, na imagem da direita da figura 2.4) e o sistema de deslocação desta, nas três direcções cartesianas (figura 2.6): um cilindro para cada direcção, com um quarto a definir o

avanço máximo em “y” (conseguindo-se assim variar o local de colocação do casquilho, para modelos diferentes de bombas) e um quinto que acciona a pinça em si. No local de alimentação de casquilhos no sistema existe um cilindro que activa um individualizador, de modo a separar o último casquilho dos restantes, para ser mais fácil a sua manipulação pela pinça de “*pick and place*”. A plataforma onde são posicionados os casquilhos (pequena plataforma com quatro elementos cilíndricos, na figura 2.4) é formada pelos locais de colocação destes e um cilindro que a posiciona ou no local de abastecimento de casquilhos ou no local para a prensagem destes. Existem três locais de colocação de casquilhos: o primeiro para um casquilho comum a todos os modelos de bombas, a nível posicional, e outros dois para modelos diferentes. Apenas um destes dois últimos locais consegue receber um casquilho a dada altura, existindo um cilindro que controla a posição destes. Já o sistema de prensagem é formado por um cilindro que faz engatar um elemento móvel no topo do corpo da bomba, fixando-o no local, e a prensa hidráulica que introduz os casquilhos. Para o trinco do corpo não desengatar, com a força aplicada pela prensa, dois cilindros activam dois elementos móveis que bloqueiam esse componente, impedindo que este se movimente verticalmente. Devido à diversidade de modelos a montar, o trinco do corpo da bomba tem dois perfis, um para o modelo Hxx e o outro para os modelos M9T e R9M. A selecção destes perfis é feita através da movimentação deste bloqueador horizontalmente por um último cilindro. Um controlador, mencionado anteriormente, é utilizado para posicionar a prensa ao longo de um eixo equivalente ao eixo “yy” da pinça de “*pick and place*”, de modo a permitir a prensagem do casquilho comum a todos os modelos e de seguida o outro casquilho, obrigando ao reposicionamento da prensa entre cada prensagem.

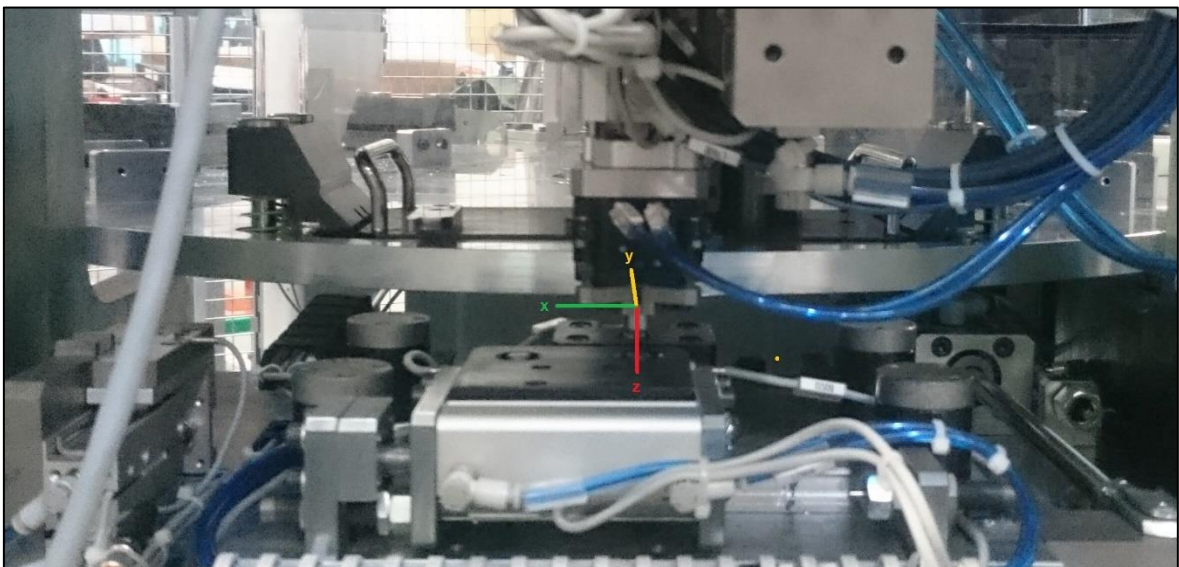


Figura 2.6: Sistema de coordenadas da pinça de “*pick and place*” de casquilhos.

A introdução de casquilhos e a prensagem funcionam de modo quase independentemente, sendo o elemento de união a plataforma de transporte dos casquilhos. Os passos necessários à apanha e colocação dos dois casquilhos nesta podem ser descritos pela seguinte sequência:

1. Avanço da pinça no eixo “xx”, ficando posicionada acima do individualizador de casquilhos. Ao mesmo tempo, avanço do individualizador de casquilhos;
2. Avanço em “yy” da pinça;
3. Avanço em “zz” da pinça (fazendo com que ela desça, ao encontro do casquilho);
4. Fecho da pinça, agarrando o casquilho;
5. Recuo em “zz”, seguido do recuo em “yy” e em “xx”;
6. Recuo do individualizador, para preparar outro casquilho para individualização;
7. Avanço em “yy” da pinça, ao encontro do local de colocação do casquilho comum;
8. Avanço em “zz” da pinça;
9. Abertura da pinça, libertando o casquilho;
10. Recuo em “zz” da pinça, seguido de recuo em “yy”;
11. Repetição dos passos “1” a “6” para apanha de segundo casquilho;
12. Avanço em “zz” da pinça”;
13. Abertura da pinça para libertar o segundo casquilho na posição não-comum, seguido de recuo em “zz”.

Já a prensagem dos casquilhos propriamente dita é realizada seguindo a seguinte sequência de operações, tendo em conta que os dois componentes se encontram na plataforma:

1. Avanço da plataforma, posicionando-se directamente abaixo do corpo da bomba de óleo;
2. Avanço do trinco do corpo;
3. Bloqueio do trinco do corpo;
4. Posicionamento da prensa hidráulica, alinhando-a com o casquilho comum;
5. Avanço da prensa, realizando a prensagem do casquilho;
6. Recuo da prensa hidráulica, concluída a operação;
7. Reposicionamento da prensa para o segundo casquilho e nova prensagem;

8. Desbloqueio do trinco;
9. Desbloqueio do corpo da bomba;
10. Recuo da plataforma, para receber novos casquilhos.

Tanto num sistema como noutro, a troca de posição de certos elementos para realizar operações em modelos diferentes é realizada no início de cada sequência (um passo “0”, diga-se), excluindo o segundo posicionamento da prensa, que é realizado no passo “7”, variando apenas a distância percorrida.

2.3. Posto de Montagem de Válvula

Esta secção é de funcionamento mais simples, contando com apenas quatro cilindros pneumáticos, sendo dois desses para troca da posição das ferramentas para montagem de modelos diferentes, e uma válvula que dá origem a vácuo na ferramenta na ponta da prensa eléctrica. O sistema de verificação utiliza dois cilindros para posicionamento da ferramenta de verificação de componentes que formam a válvula, ou para retorno da ferramenta para a posição de repouso. O terceiro reposiciona este sistema para diferentes modelos de bomba a produzir, com o quarto a realizar o mesmo trabalho, mas relativamente à posição da prensa. Esta secção conta também, como dito anteriormente, com intervenção por parte de um operador, para realizar certas operações. A passagem entre o período de montagem manual e montagem automática é dada pelo próprio operador, ao carregar num botão.

A ferramenta de verificação é formada por um sensor indutivo e um potenciómetro. Ao fazer a verificação, caso o pistão e cavilha estejam correctamente inseridos, a ponta da ferramenta recuará, sendo esse recuo detectado pelo sensor. Ao mesmo tempo, o comprimento da mola é verificado pelo potenciómetro, através de um segundo elemento móvel na ferramenta. Este sistema garante que não são utilizados componentes relativo a modelos M9T em bombas Hxx, por exemplo, dado que os comprimentos das molas são diferentes, assim como a distância a que recua a ponta de verificação. Ferramentas iguais, mas com sensores calibrados para distâncias de recuo diferentes são, portanto, utilizados e trocados pelo utilizador. Na figura 2.7 pode-se observar o sistema de verificação posicionado acima do local da válvula no corpo da bomba, a realizar uma aproximação para verificação dos componentes. É possível ver também a mola em posição correcta.

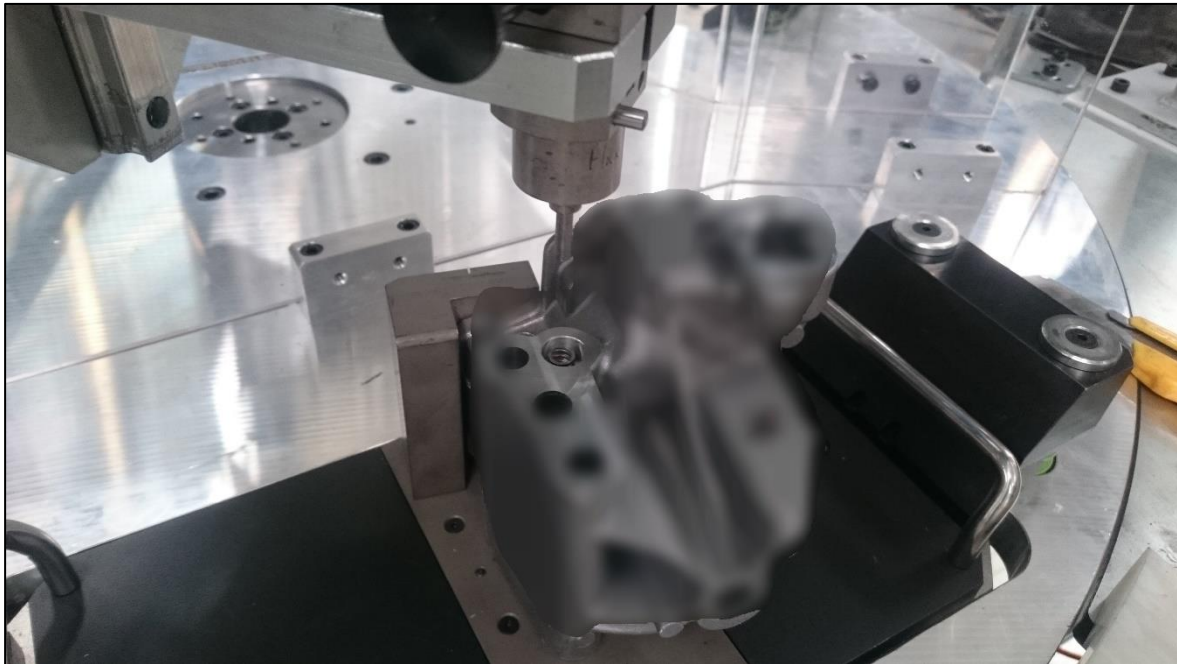


Figura 2.7: Operação de verificação de componentes da válvula num modelo M9T.

Para a prensagem, é activado vácuo na ferramenta da prensa, permitindo o encaixe do tampão da válvula nesta, sendo que a presença deste é detectado através de um vacuostato. O vácuo é apenas activado, caso se vá montar uma válvula num modelo M9T ou R9M, após verificação da existência de filtro no pistão (pelo operador). Dependendo do modelo de bomba a produzir, a distância e força aplicada na prensagem variam. A ferramenta da prensa é também ela mudada pelo operador. Estas ferramentas de prensagem, assim como as ferramentas de verificação, são armazenadas num local próprio no posto de trabalho dotado de sensores que verificam quais as ferramentas em uso. Isto garante que a operação não é realizada caso as ferramentas estejam trocadas – por exemplo, ao tentar-se utilizar uma ferramenta para o modelo R9M numa prensagem de um modelo Hxx.

A sequência de passos para a operação de montagem da válvula no corpo da bomba de óleo é a seguinte:

1. Reposicionamento automático do sistema de verificação e da prensa eléctrica, em caso de mudança de modelo a montar;
2. Verificação da existência de filtro no pistão, no caso de modelos M9T e R9M;
3. Activação do vácuo na ferramenta da prensa;
4. Colocação, pelo operador, do pistão, mola, cavilha e tampão nos locais correctos;
5. Validação do início das operações automáticas;
6. Verificação da existência e correcta colocação de pistão, cavilha e mola;

7. Em caso de resultado “mau”, repetição dos passos “4” a 6”;
8. Prensagem do tampão da válvula no corpo da bomba de óleo;
9. Descarga do corpo da bomba do posto de trabalho e mudança das ferramentas de verificação e prensagem e bloco de posicionamento, caso o modelo de bomba seguinte seja diferente do anterior.

Pode-se ver uma fotografia da secção de montagem da válvula na figura 2.8, com a ponta da prensa com a sua ferramenta ao centro, o sistema de verificação de componentes ao centro, à esquerda, o bloco de posicionamento, onde o corpo da bomba de óleo estaria posicionado, em baixo e o local de armazenamento de ferramentas em cima, à direita, mostrando uma ferramenta para a prensa.

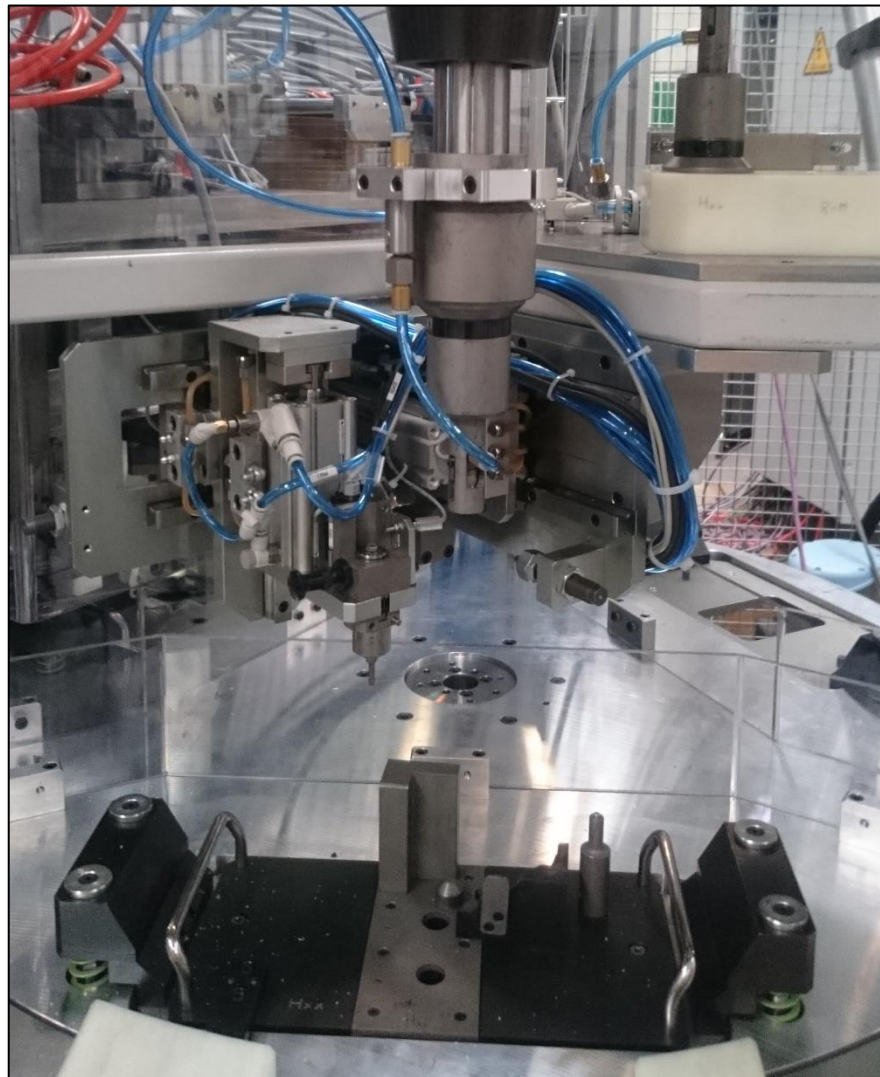


Figura 2.8: Secção do OP25 para montagem de válvula no corpo da bomba de óleo.

2.4. Sistemas de Segurança

Os sistemas de segurança implementados neste posto de trabalho destinam-se não só a proteger indivíduos da máquina, mas também a própria máquina. Tem-se os exemplos de, em primeiro, se ter que impedir que um operador tenha acesso a zonas do posto onde elementos móveis deste se estejam a movimentar, para prevenir acidentes físicos e, em segundo, se garantir que elementos móveis não chocam entre si, o que poderia originar quebras mecânicas e obrigar à paragem para manutenção do OP25.

Existem três tipos de sistemas de segurança implementados: mecânicos, eléctricos e programáticos. Os primeiros são constituídos por barreiras físicas como divisórias de rede, painéis de policarbonato e portas, que impedem o acesso a certas zonas da máquina, quando esta está em funcionamento. Podem-se observar exemplos de policarbonato nas figuras 2.7 e 2.8, impedindo o acesso do operador no posto de montagem de válvula a outras zonas da máquina.

Quanto a seguranças eléctricas, temos todos os dispositivos conectados aos relés de segurança existentes no quadro eléctrico. Estes incluem a paragem de emergência, a fechadura de acesso ao interior da ilha do posto de trabalho (ao interior da área delimitada pelas divisórias de rede), o fecho magnético da porta de acesso à secção de inserção dos casquilhos e as barreiras imateriais colocadas no posto de montagem de válvulas, que detectam quando o operador tem partes do corpo no interior da secção. Se algum destes dispositivos é accionado, os relés de segurança disparam, cortando a alimentação a todos os dispositivos capazes de movimentar partes da máquina: o robô, válvulas que accionam cilindros, a prensa eléctrica, o controlador de posição da prensa hidráulica e o indexador circular que movimenta o prato rotativo. No caso das barreiras imateriais, apenas os elementos da secção associadas são parados. Isto permite que uma bomba que esteja a ser trabalhada na secção dos casquilhos não veja a sua montagem interrompida de cada vez que o operador necessita colocar componentes nos corpos das bombas.

Finalmente, as seguranças programadas baseiam-se no programa do autómato. Neste, desenvolveram-se funções que não permitem que certos elementos possam ser accionados simultaneamente, para não chocarem.