



Instituto Politécnico de Tomar

**Escola Superior de Tecnológica de Tomar**

**Micael Gameiro Rodrigues**

**Desenvolvimento de sistemas de  
automação e de supervisão para  
máquinas industriais**

Relatório de Estágio

Orientado por:

Pedro Daniel Frazão Correia

Relatório de Estágio apresentada ao Instituto Politécnico de Tomar para  
cumprimento dos requisitos necessários à obtenção do grau de Mestre  
(Mestrado em Engenharia Eletrotécnica, Especialidade em Controlo e Eletrónica  
Industrial)

**Tomar, setembro de 2019**



Dedico este trabalho aos meus Pais,



## RESUMO

---

Este relatório de estágio foi desenvolvido no âmbito da unidade curricular de Projeto do 2º ano do Mestrado em Engenharia Eletrotécnica – Especialidade em Controlo e Eletrónica Industrial e traduz o trabalho decorrente do estágio curricular de nove meses na realizado na empresa Geralógica, LDA.

O presente relatório apresenta o desenvolvimento de dois projetos: i) sistema para controlo e supervisão de um processo de rotomoldagem; ii) divisor angular para uma máquina CNC de dois eixos (do inglês, *Computer Numerical Control*).

A máquina de rotomoldagem permite a realização de moldagem rotacional, que consiste num processo mecânico para fabricação de peças plásticas de grandes dimensões, onde a matéria-prima não sólida é moldada até obtenção do produto final. O sistema desenvolvido tem várias valências desde a monitorização e controlo do processo em tempo real, a possibilidade da execução de várias receitas, para diferentes moldes e o desenvolvimento de relatórios de produção e alarmes.

O divisor angular é concebido para uma máquina CNC de dois eixos, permitindo um aumento a sua capacidade de resposta, adicionando a um custo reduzido um terceiro eixo. Este eixo é rotativo, permitindo de forma automática a rotação de peças durante o processo de maquinação. Neste projeto, foi implementado o esquema elétrico do módulo, o programa de controlo e o respetivo interface.

Os projetos desenvolvidos ficaram em fase de produção e testes nos respetivos clientes.

**Palavras-chave:** SCADA, Moldagem Rotacional, Automação Industrial, CNC, Divisor Angular.



# ABSTRACT

---

This report was developed within the scope of the 2nd year Project Curricular Unit of the master's degree in Electrical Engineering - Specialisation in Control and Industrial Electronics describing the nine-month curricular internship work carried out at Geralógica, LDA.

This report describes the development of two projects: i) a control and supervision system for a rotational molding process; ii) an angular divider for a two axis CNC machine (Computer Numerical Control) including the control and supervision tasks.

The rotational machine allows rotational molding consisting in a mechanical process for the manufacturing of large plastic parts, where the non-solid raw material is molded until the final product is obtained. The developed system has several advantages, such as, the real-time monitoring and control process, several receipts configuration for different molds, production reports and alarms generation.

The angular divider is designed for a second axis CNC machine, increasing the responsiveness by adding a third axis at a reduced cost. This is a rotary axis, allowing automatically rotate the parts during the machining process. In this project, the electrical scheme of the module, the control program and the respective interface were implemented.

The developed projects have been in production and testing phase at the respective clients.

**Keywords:** SCADA, Rotational Molding, Industrial Automation, CNC, Angular Divider



## AGRADECIMENTOS

---

Esta secção é dedicada a todos aqueles que me apoiaram ao longo destes anos, que proporciona a conclusão desta etapa.

Em primeiro, quero agradecer a toda minha família em especial aos meus pais, que sempre me apoiaram e motivaram para chegar a este patamar. À minha irmã que teve sempre presente com as suas palavras e conselhos encorajadores.

À minha namorada que sempre teve presente, para um apoio emocional muito grande.

Ao orientador Doutor Pedro Daniel Frazão Correia, pelo seu apoio, conselhos, empenho e pela disponibilidade com que acompanhou este relatório de estágio.

A empresa onde realizei o estágio, Geralógica, em especial ao Eng. João Capitão, Paulo Silva e ao Eng. Paulo Martins pela oportunidade e pela ajuda que facultaram ao longo de todo o estágio.

.



# Índice

|   |      |
|---|------|
| Índice .....  | XI   |
| Índice de figuras .....                                   | XV   |
| Índice de tabelas .....                                   | XVII |
| Lista de abreviaturas e siglas .....                      | XIX  |
| Lista de símbolos .....                                   | XXI  |
| 1. Introdução .....                                       | 22   |
| 1.1. Entidade Acolhedora .....                            | 23   |
| 2. Estado de arte .....                                   | 24   |
| 2.1. Evolução dos sistemas de Automação .....             | 25   |
| 2.2. Evolução da arquitetura do sistema de controlo ..... | 26   |
| 2.3. Classificação das redes industriais .....            | 27   |
| 2.4. Scada .....  | 29   |
| 2.4.1. Arquitetura dos sistemas de comunicação .....      | 30   |
| 2.4.1.1. Modelo OSI .....                                 | 30   |
| 2.4.1.2. Norma IEC 60870-5 .....                          | 31   |
| 2.4.2. Arquitetura dos sistemas SCADA .....               | 32   |
| 2.4.2.1. Monolithic SCADA Systems .....                   | 32   |
| 2.4.2.2. Distributed SCADA Systems .....                  | 33   |
| 2.4.2.3. Networked SCADA Systems .....                    | 35   |
| 2.4.2.4. Internet of Things .....                         | 36   |
| 2.4.3. Protocolos de comunicação .....                    | 38   |
| 2.4.3.1. Protocolo Profibus .....                         | 38   |
| 2.4.3.2. Protocolo Ethernet .....                         | 39   |
| 2.4.3.3. Protocolo de Modbus .....                        | 40   |
| 2.4.3.3.1. Modos de operação do protocolo Modbus .....    | 42   |
| 2.4.4. Componentes do Sistema SCADA .....                 | 43   |
| 2.4.4.1. Unidades de Terminal Remoto (RTU) .....          | 44   |
| 2.4.4.2. Controlador lógico programável (PLC) .....       | 44   |
| 2.4.4.3. Interface Homem-Máquina (HMI) .....              | 45   |
| 2.4.4.4. Base de dados .....                              | 45   |
| 2.4.4.5. Supervisão .....                                 | 45   |
| 2.4.5. Software SCADA: Movicon .....                      | 45   |
| 2.4.6. Comunicação com a base de dados .....              | 47   |

|          |   |    |
|----------|---|----|
| 3.       | Processo de Rotomoldagem .....                      | 48 |
| 3.1.     | Rotomoldagem .....                                  | 49 |
| 3.2.     | Tipos de materiais aplicados na rotomoldagem .....  | 51 |
| 3.2.1.   | Polietileno de Baixa Densidade (LDPE).....          | 51 |
| 3.2.2.   | Polietileno de Alta Densidade (HDPE).....           | 52 |
| 3.2.3.   | Polietileno de Média Densidade (MDPE).....          | 52 |
| 3.2.4.   | Polietileno linear de baixa densidade (LLDPE) ..... | 52 |
| 3.2.5.   | Etileno Vinil Acetato (EVA) .....                   | 53 |
| 3.3.     | Tipos de máquinas de moldagem rotacional.....       | 54 |
| 3.3.1.   | Rock-and-Roll .....                                 | 54 |
| 3.3.2.   | Clamshell.....                                      | 55 |
| 3.3.3.   | Shuttle Machines .....                              | 55 |
| 3.3.4.   | Fixed-Arm Carousel Machine .....                    | 56 |
| 3.3.5.   | Independent-Arm Machine.....                        | 57 |
| 4.       | Implementação da máquina de Rotomoldagem .....      | 58 |
| 4.1.     | Funcionamento da máquina.....                       | 59 |
| 4.2.     | Variadores de velocidade.....                       | 61 |
| 4.3.     | Minicomputador: Advantech UNO-2362G .....           | 62 |
| 4.4.     | LCD: FPM-5171G.....                                 | 63 |
| 4.5.     | Autómato: NX1P2.....                                | 64 |
| 4.5.1.   | Programa de controlo.....                           | 67 |
| 4.5.1.1. | Geral .....   | 67 |
| 4.5.1.2. | Gestão de Programas .....                           | 68 |
| 4.5.1.3. | Forno .....   | 68 |
| 4.5.1.4. | Carros .....  | 69 |
| 4.5.1.5. | Balanceamento.....                                  | 70 |
| 4.5.1.6. | RS485 .....   | 71 |
| 4.5.1.7. | Alarmes .....                                       | 73 |
| 4.6.     | Autómato de segurança .....                         | 74 |
| 4.6.1.   | Programação do autómato de segurança .....          | 79 |
| 4.7.     | Sistema de supervisão.....                          | 80 |
| 4.7.1.   | Página principal: Forno.....                        | 80 |
| 4.7.2.   | Registo de produção .....                           | 81 |
| 4.7.2.1. | Materiais .....                                     | 82 |
| 4.7.2.2. | Moldes .....  | 83 |
| 4.7.2.3. | Classificação.....                                  | 83 |

|          |   |     |
|----------|---|-----|
| 4.7.3.   | Relatório de produção.....                    | 84  |
| 4.7.3.1. | Script geral.....                             | 87  |
| 4.7.4.   | Gestão das Receitas .....                     | 88  |
| 4.7.4.1. | Gravação da Receita .....                     | 90  |
| 4.7.4.2. | Eliminar uma receita.....                     | 92  |
| 4.7.4.3. | Cancelar edição de receita .....              | 93  |
| 4.7.4.4. | Upload e Download .....                       | 93  |
| 4.7.4.5. | Configurações avançadas.....                  | 94  |
| 4.7.5.   | Alarmes .....                                 | 94  |
| 4.7.6.   | Gestão do Balanceamento.....                  | 98  |
| 5.       | Implementação do divisor angular.....         | 99  |
| 5.1.     | Controlo numérico computadorizado (CNC) ..... | 100 |
| 5.2.     | Divisor angular .....                         | 101 |
| 5.3.     | Motor 60HSE3N-D25 .....                       | 103 |
| 5.4.     | Travão eletromagnético .....                  | 104 |
| 5.5.     | Servo Drive .....                             | 104 |
| 5.6.     | Autómato: CP1E.....                           | 106 |
| 5.6.1.   | Programa de controlo.....                     | 107 |
| 5.6.1.1. | Geral .....                                   | 107 |
| 5.6.1.2. | Modo de furo .....                            | 108 |
| 5.6.1.3. | Modo Tabela.....                              | 109 |
| 5.6.1.4. | Travão.....                                   | 110 |
| 5.6.1.5. | Alarmes .....                                 | 111 |
| 5.6.1.6. | Saídas .....                                  | 111 |
| 5.7.     | Consola: NB7W .....                           | 112 |
| 5.7.1.   | Sistema de supervisão.....                    | 112 |
| 6.       | Conclusão.....                                | 115 |
|          | Bibliografia .....                            | 117 |
|          | Anexos .....                                  | 119 |



## Índice de figuras

|   |    |
|---|----|
| Figura 1. Pirâmide de sistemas de Automação .....                           | 27 |
| Figura 2. Exemplo de uma rede SCADA: <i>Monolithic</i> . .....              | 33 |
| Figura 3. Exemplo de uma rede SCADA: <i>Distributed</i> .....               | 34 |
| Figura 4. Exemplo de uma rede SCADA: <i>Networked</i> .....                 | 36 |
| Figura 5. Exemplo de uma implementação do IOT. ....                         | 37 |
| Figura 6. Tipo mensagem: <i>Unicast</i> .....                               | 41 |
| Figura 7. Tipo mensagem: <i>Broadcast</i> .....                             | 41 |
| Figura 8. Estrutura de um <i>frame</i> Modbus em RTU .....                  | 42 |
| Figura 9. Estrutura de uma <i>frame</i> Modbus em ASCII .....               | 43 |
| Figura 10. Rede de um Sistema de SCADA (6) .....                            | 44 |
| Figura 11. Arquitetura geral da integração de ADO's .....                   | 47 |
| Figura 12. Reservatório Vertical em Polietileno.....                        | 49 |
| Figura 13. Reservatório subterrâneo em polietileno .....                    | 49 |
| Figura 14. Processo de rotomoldagem (24).....                               | 50 |
| Figura 15. Exemplo da ramificação dos polímeros (24).....                   | 53 |
| Figura 16. Máquina de <i>Rock-and-roll</i> .....                            | 54 |
| Figura 17. Máquina de <i>Clamshell</i> .....                                | 55 |
| Figura 18. Máquina <i>Shuttle</i> (25).....                                 | 56 |
| Figura 19. Máquina de carrossel .....                                       | 57 |
| Figura 20. Máquina de Rotomoldagem .....                                    | 59 |
| Figura 21. Estufa .....   | 59 |
| Figura 22. Carro 1 .....  | 60 |
| Figura 23. Componentes de controlo da máquina. ....                         | 60 |
| Figura 24. Esquema de ligação do variador A1000.....                        | 62 |
| Figura 25. Advantech UNO-2362G.....   | 63 |
| Figura 26. FPM-5171G .....  | 63 |
| Figura 27. Autómato: NX1P2. ....  | 64 |
| Figura 28. Comunicação entre o módulo CIF e a sonda de temperatura.....     | 66 |
| Figura 29. Seções do programa de controlo rotomoldagem.....                 | 67 |
| Figura 30. Fluxograma da gestão dos carros .....                            | 70 |
| Figura 31. Especificação do módulo de comunicação. ....                     | 71 |
| Figura 32. Ativação de cada nó. ....  | 71 |
| Figura 33. Exemplo de uma leitura do variador .....                         | 72 |
| Figura 34. Registos utilizados para monitorização dos variadores (26) ..... | 72 |
| Figura 35. Exemplo de acesso a um registo de controlo.....                  | 72 |
| Figura 36. Código de comando da sonda de temperatura (27).....              | 73 |
| Figura 37. Exemplo de um alarme.....  | 73 |
| Figura 38. Gestão dos alarmes.....  | 74 |
| Figura 39. Gestão <i>acknowledge</i> .....                                  | 74 |
| Figura 40. Autómato: G9SP-N10S.....   | 75 |
| Figura 41. Esquema de entradas no autómato de segurança. ....               | 77 |
| Figura 42. Esquema de Ligação das cortinas de luz. ....                     | 78 |

|  |     |
|--|-----|
| Figura 43. Programação das Barreiras de segurança.....       | 79  |
| Figura 44. Programação do botão de emergência.....           | 79  |
| Figura 45. Ecrã da página do Forno.....                      | 80  |
| Figura 46. Página de registo de produção.....                | 82  |
| Figura 47. Página Moldes.....                                | 83  |
| Figura 48. Página Classificação.....                         | 84  |
| Figura 49. Página relatório de produção.....                 | 85  |
| Figura 50. Concatenação da data.....                         | 85  |
| Figura 51. Instrução <i>Select Count</i> .....               | 85  |
| Figura 52. Exemplo de uma filtragem.....                     | 86  |
| Figura 53. Tabela do Relatório de Produção.....              | 86  |
| Figura 54. Exemplo da Criação da tabela da Auxiliar.....     | 87  |
| Figura 55. <i>Template</i> do relatório de produção.....     | 87  |
| Figura 56. Definir um evento.....                            | 88  |
| Figura 57. Gestão das Receitas.....                          | 89  |
| Figura 58. Fluxograma de gravação da receita.....            | 90  |
| Figura 59. Escrita no ficheiro texto.....                    | 91  |
| Figura 60. <i>Pop-up</i> : Colocação do nome da receita..... | 91  |
| Figura 61. Fluxograma de eliminar receita.....               | 92  |
| Figura 62. Janela das configurações avançadas.....           | 94  |
| Figura 63. <i>Array</i> dos Alarmes do autómato.....         | 95  |
| Figura 64. Exemplo de Alarmes.....                           | 95  |
| Figura 65. Exemplo da aplicação do " <i>Threshold</i> "..... | 96  |
| Figura 66. Exemplo de uma listagem de alarme.....            | 96  |
| Figura 67. Exemplo do <i>Log Window</i> .....                | 97  |
| Figura 68. Gestão de Balanceamento.....                      | 98  |
| Figura 69. CNC de um eixo.....                               | 100 |
| Figura 70. CNC de 3 eixos.....                               | 101 |
| Figura 71. Divisor Angular.....                              | 101 |
| Figura 72. Modos de Funcionamento.....                       | 102 |
| Figura 73. Quadro elétrico do divisor angular.....           | 102 |
| Figura 74. Especificações do motor (31).....                 | 103 |
| Figura 75. Curva de binário. (31).....                       | 103 |
| Figura 76. Constituição de um travão eletromagnético.....    | 104 |
| Figura 77. <i>Servo drive</i> .....                          | 105 |
| Figura 78. Malha de realimentação dos Servos Drive (32)..... | 105 |
| Figura 79. Seções do programa do divisor angular.....        | 107 |
| Figura 80. Exemplo da função PLS2.....                       | 108 |
| Figura 81. Configuração do pulso.....                        | 109 |
| Figura 82. Fluxograma do funcionamento do modo tabela.....   | 110 |
| Figura 83. Saídas do Divisor Angular.....                    | 111 |
| Figura 84. Consola aplicada no processo.....                 | 112 |
| Figura 85. Configuração entre PLC e HMI no NB-Designer.....  | 113 |
| Figura 86. Atribuir um bit a um botão.....                   | 113 |
| Figura 87. <i>Template</i> da página principal.....          | 114 |
| Figura 88. Modo de tabela de pontos.....                     | 114 |

## Índice de tabelas

|  |     |
|--|-----|
| Tabela 1. Modelo OSI.....                              | 30  |
| Tabela 2. Característica do NX1P2. ....                | 65  |
| Tabela 3. Especificações do autômato: G9SP-N10 S ..... | 76  |
| Tabela 4. Entradas do autômato: G9SP-N10 S.....        | 77  |
| Tabela 5. Saídas do autômato: G9SP-N10 S .....         | 78  |
| Tabela 6. Entradas do autômato do Divisor Angular..... | 106 |
| Tabela 7. Saídas do autômato do Divisor Angular .....  | 106 |



## **Lista de abreviaturas e siglas**

ASCII - *American standard code for information interchange;*

ADO- *Microsoft ActiveX Data Objects;*

CIM - *Computer Integrated Manufacturing;*

CNC - *Computer Numeric Control;*

DBMS - *Database Management System;*

DCS - *Distributed Control Systems;*

DDC - *Direct Digital Control;*

DLR - *Device Level Ring;*

EVA - *Etileno Vinil Acetato;*

FCS - *Fieldbus Control System;*

HDPE - *Polietileno de Alta Densidade;*

HMI - *Interface Homem-Máquina;*

IoT- *Internet das coisas;*

IP - *Internet Protocol;*

ISSO - *Open System Interconnection;*

LAN - *Rede de área local;*

LDPE - *Polietileno de Baixa Densidade;*

LLDPE - *Polietileno linear de baixa densidade;*

M2M - *Comunicações máquina a máquina;*

MDPE - *Polietileno de Média Densidade;*

MODICON - *Modular Digital Controller;*

MQTT - *Message Queuing Telemetry Transport;*

ODBC - *Open Database Connectivity*;

OLE - *Object Linking and Embedding*;

PC - Computador pessoal;

PLC - Controlador lógico programável;

RTU - Unidades de Terminal Remoto;

SCADA - Sistemas de Supervisão e Aquisição de Dados;

SQL - Linguagem de Consulta Estruturada;

SSMS - *SQL Server Management Studio*;

UDP - *User datagram protocol*;

VBA - *Visual Basic* para Aplicações;

WAN - *Wide Area Networks*;

XML - *Extensive markup language*.

## Lista de símbolos

5G- Quinta geração de comunicações móveis;

A- Ampere, unidade de intensidade de corrente elétrica;

b- Gigabyte, unidade de medida de armazenamento;

Bps- Bits transmitidos por segundo, unidade de transmissão de dados;

°C- Unidade de medida de temperatura, na escala Celsius;

G- Aceleração da gravidade;

Grms- Valor eficaz da amplitude das vibrações, medida em aceleração;

Hz - Hertz, unidade de medida para a frequência;

$m^3$ - Metro cúbico, unidade de medida de volume;

V- Volt, unidade de tensão elétrica.

## 1. Introdução

O presente relatório de estágio descreve o trabalho realizado durante o estágio curricular na empresa Geralógica.

O objetivo deste estágio foi o desenvolvimento de vários processos de automação, entre os quais um sistema de supervisão para um processo de rotomoldagem, assim como um divisor angular.

A rotomoldagem rotacional é um processo mecânico para fabricação de peças plásticas de grandes dimensões, onde a matéria-prima não sólida é moldada até obtenção do produto final. Este processo é o único viável para a produção de objetos com grandes dimensões ( $> 2 m^3$ ) em peças únicas. Este método é conhecido pela fabricação de tanques, mas também é utilizada em brinquedos e embarcações. Este processo de rotomoldagem foi integralmente construído na empresa, desde os componentes mecânicos, ao desenvolvimento dos quadros elétricos, à eletrificação do processo assim como à componente de automação do processo.

O divisor angular é um acessório de uma máquina CNC (do inglês, *Computer Numeric Control*) que permite aumentar a capacidade de resposta adicionando um terceiro eixo a um custo reduzido. Este eixo é rotativo permite a rotação de peças durante o processo de maquinação.

Os projetos mencionados foram desenvolvidos para a indústria, com uma grande preocupação para obter bons resultados ao nível da sua durabilidade, fiabilidade e robustez, assim como da simplicidade de funcionamento.

Em termos de organização, este relatório de estágio encontra-se dividido em 6 capítulos. O capítulo 1 descreve os objetivos do estágio e apresenta a empresa acolhedora. O capítulo 2 apresenta um estudo da evolução histórica dos sistemas de automação e de supervisão, assim como os tipos de arquiteturas e componentes num sistema de supervisão. Descreve também as arquiteturas dos sistemas de comunicação e os protocolos de comunicação aplicados nos projetos desenvolvidos. No capítulo 3, é explicado o processo de rotomoldagem, os materiais utilizados e ainda as máquinas existentes no mercado. O capítulo 4 aborda a máquina de rotomoldagem, desde os componentes físicos até à explicação da programação executada no projeto. No capítulo

5 é abordado o divisor angular onde são descritos os objetivos deste processo, os modos de funcionamento, bem como a programação do modelo. Por último, no capítulo 6 é apresentada a conclusão.

## **1.1. Entidade Acolhedora**

A Geralógica, Lda encontra-se sediada em Amor, no distrito de Leiria. A empresa foi fundada em maio de 2006 por Paulo Sérgio Ferreira Martins e Paulo Mendes Silva.

Esta iniciou a sua atividade com a reparação de pequenos equipamentos elétricos. Com o seu crescimento expandiu a atividade para a remodelação de equipamentos automatizados.

No ano 2008 iniciou-se uma parceria com a empresa Leirimetal, permitindo um grande crescimento da empresa e a sua internacionalização com a realização de diversos trabalhos na Venezuela e Espanha.

Com o seu crescimento e pela necessidade do mercado, a empresa aumentou a sua área de negócio e criou um departamento de mecânica com o intuito de criar equipamentos feitos à medida dos clientes.

As áreas de negócio atuais vão desde a manutenção industrial, à remodelação de máquinas e criação de máquinas à medida do cliente, incluindo os componentes mecânicos, elétricos e de automação.

Neste momento a empresa conta com uma equipa multidisciplinar, tem cerca de dez elementos, entre engenheiros eletrotécnicos, engenheiros mecânicos, serralheiros e operadores de CNC.

## 2. Estado de arte

Neste capítulo é apresentado um resumo da evolução dos sistemas de automação industrial e os respetivos sistemas de controlo. Em particular é apresentada a evolução dos sistemas de supervisão e de aquisição de dados, vulgarmente denominados como sistemas SCADA (do inglês, *Supervisory Control and Data Acquisition*). Aborda-se também as diferentes gerações de redes de dados e dos protocolos de comunicação mais utilizados neste tipo de aplicações. Por fim é apresentado o *software* de SCADA usado no processo de rotomoldagem.

## 2.1. Evolução dos sistemas de Automação

Com o aparecimento dos sistemas automáticos de apoio à produção industrial ou como vulgarmente se denomina, dos sistemas de automação industrial, houve um aumento da produtividade, e em consequência, a melhoria das condições de vida. O processo de automação, que se alarga às diferentes áreas de atividade, tem um papel fundamental no avanço da engenharia e ciência. Este foi especialmente importante na corrida espacial e da aviação comercial. A automação permite que as tarefas repetitivas, tradicionalmente realizadas por pessoas, sejam realizadas por máquinas, permitindo economizar energia por meio de ganhos de eficiência, reduzir a quantidade de materiais necessários permitindo a melhoria da qualidade, precisão e previsibilidade dos processos. Paralelamente, garante a melhoria da segurança das pessoas e bens ao retirar as pessoas de condições inseguras ou perigosas (1).

Um dos maiores marcos no desenvolvimento da automação foi a criação da máquina a vapor, determinando o período da primeira revolução industrial que começou algures entre os anos de 1760 até 1840 em Inglaterra. Esta permitiu o aumento da produção de artigos manufaturados. No século seguinte, com as novas fontes de energia e a substituição do ferro pelo aço, assistiu-se ao desenvolvimento das indústrias na Europa e nos Estados Unidos (2).

No início do século XX, com Henry Ford, existiu uma revolução em todo o pensamento da indústria contemporânea, propagando-se até os dias de hoje. Foi introduzida a linha de montagem, permitindo um aumento da produtividade e a redução de tempo, reduzindo assim os custos de produção (2).

Com a segunda Guerra Mundial e posteriormente a corrida espacial registada na segunda metade do século XX, existiram desenvolvimentos tecnológicos importantes, desde a eletrónica às redes de comunicação. Neste contexto, foram desenvolvidos e aperfeiçoados os computadores digitais e microcontroladores que viriam mais tarde a dar lugar ao surgimento dos primeiros Controladores Industriais, Controladores Lógicos Programáveis ou PLC (do inglês, *Programable Logic Controller*).

Estes desenvolvimentos permitiram um aumento gradual da quantidade de trabalho, contribuindo para aumentar simultaneamente a eficiência na produção industrial, através

da redução dos consumos de energia e do desperdício de matérias-primas, assim como o aumento da qualidade dos produtos finais.

Os primeiros Controladores Lógicos Programáveis, foram desenvolvidos em 1968, pela empresa *BedFord Association*, em BedFord nos Estados Unidos. Esta foi contratada para desenvolver um dispositivo eletrónico que substituísse os relés. O modelo deste autómato foi o MODICON (do inglês, *Modular Digital Controller*), modelo 084. A sua primeira aplicação foi na empresa *Bryant Chuck and Grinder* em Springfield, Vermont, Estados Unidos da América (2).

## **2.2. Evolução da arquitetura do sistema de controlo**

No ano de 1960 iniciou-se a substituição de sistemas pneumáticos por sistemas elétricos. Estes sistemas de instrumentação utilizavam um sinal contínuo, com uma gama de corrente de 4 até 20mA. Neste caso, um computador central recebe informações dos sensores, enviando sinais de comando para os atuadores de acordo com a leitura efetuada. Estes sistemas designavam-se por controlo digital direto, DDC (do inglês, *Direct Digital Control*) (3).

As vantagens deste tipo de sistemas eram a recolha e processamento da informação dos sensores permitindo um controlo mais eficiente. No entanto, as unidades de controlo eram lentas, pouco fiáveis e com um custo elevado.

Os sistemas DDC tiveram um grande desenvolvimento com o aparecimento dos minicomputadores, estimando-se que houvesse cerca de 5000 sistemas DDC em 1970 e perto de 50 000 em 1975 (3).

No ano de 1975, foram desenvolvidos os sistemas de controlo distribuído, DCS (do inglês, *Distributed Control Systems*). Neste caso, várias unidades de campo independentes compostas por toda a instrumentação de vários processos estavam ligadas a um microprocessador ou um autómato. Estas unidades eram ligadas através de um “bus” comum, ligadas a um computador central. Com isto, foi possível aumentar a confiabilidade reduzindo ao mesmo tempo os custos de instalação e permitindo a supervisão remota.

Com o nível de complexidade dos processos industriais a aumentar, foi necessário criar mecanismos para simplificar os processos. Um dos outros problemas associados era o problema das interferências eletromagnéticas nos sinais analógicos. Para melhorar esses problemas foi criado um sistema designado na literatura anglo-saxónica por FCS (do inglês, *Fieldbus Control System*) (3).

O FCS é um sistema que pode-se definir como uma rede digital bidirecional, utilizado para interligar os dispositivos primários de automação a um sistema integrado de automação e controlo de processo. Cada dispositivo local possui um microprocessador, o que o torna capaz de executar funções simples, como diagnósticos, controlo de processo, alarmes e possibilita a comunicação entre dispositivos do campo. Este sistema vem substituir o comando centralizado num só local.

### 2.3. Classificação das redes industriais

Atualmente existem diferentes tipos de modelos conceptuais que descrevem sistemas que coordenam um processo produtivo.

A classificação aplicada neste relatório é um modelo baseado em pirâmide, em que está estruturado em diferentes hierarquias. A criação deste modelo é baseado no modelo CIM (do inglês, *Computer Integrated Manufacturing*). Este método de organização por níveis é muito útil para o desenvolvimento de sistemas de automação integrados e sincronizados, como é ilustrado na Figura 1 (4).



Figura 1. Pirâmide de sistemas de Automação

Na primeira camada é apresentada a seção de instrumentação, onde se encontra os dispositivos de campo. Estes são responsáveis pela aquisição de toda a informação do processo e ainda pela execução de todas as ordens dadas pelos sistemas de controle. Os modos de comunicação entre este nível são tipicamente o Can, Hart, Fieldbus e Profibus DP (4).

A segunda camada é preenchida pelos sistemas de comandos sejam os autômatos, RTU (do inglês, *Remote Terminal Unit*), variadores de velocidade e entre todos os outros equipamentos usados. A comunicação pode ser feita pelos protocolos ControlNet, Ethernet, Modbus, Profibus FMS ou Profinet (4).

O nível seguinte é ocupado pelos sistemas de supervisão. Estes permitem observar, otimizar, gerar alarmes, entre outras funções. Estes sistemas tipicamente têm uma base de dados com todas as informações relativas ao processo (4).

A gestão da produção é responsável pela programação e o planeamento da produção, realização do controlo de dados da supervisão e ainda a logística da empresa. Nesta camada são aplicados sistemas com MES (do inglês, *Manufacturing Execution System*), PIMS (do inglês, *Process Information Management System*), APS (do inglês, *Advanced Planning and Scheduling*), LIMS (do inglês, *Lab Information System*), Sistemas de Manutenção (do inglês, *Maintenance Management System*) e Sistema de Gestão de Ativos (do inglês, *Asset Management System*).

O ERP, (do inglês, *Enterprise Resource Planning*). Este é responsável pela gestão dos processos de negócio que gere e integra as atividades de finanças, cadeia de fornecimento, operações, relatórios e recursos humanos da empresa (5).

Por último, os sistemas de DW (do inglês, *Data Warehousing*) estes armazenam todos os dados relativos às atividades da organização numa base de dados e os sistemas EIS (do inglês, *Executive Information Systems*), que têm como objetivo principal dar suporte à tomada de decisão (4).

## 2.4. Scada

Nos ambientes industriais a gestão dos processos é cada vez mais importante e crucial para a sua eficiência. Os sistemas de Sistemas de Supervisão e Controlo Industrial (SCADA) surgiram para atingir um grande controlo do processo.

Os constituintes dos sistemas de SCADA englobam todos os elementos de *software* e *hardware*. Estes sistemas têm como principais funções controlar processos industriais, sejam locais ou remotos, monitorizar e processar dados em tempo real, interagir com dispositivos como sensores, válvulas, bombas, motores e muito mais por meio dos *softwares* de interface homem-máquina, HMI (do inglês, *Human-Machine Interface*) (6).

Os sistemas de SCADA surgiram na década de 60 pela Westinghouse e GE, com o aparecimento dos computadores capazes de realizar funções em tempo real.

Nas décadas de 80 e 90, os sistemas de SCADA evoluíram consideravelmente devido ao aumento da capacidade de processamento dos computadores, o surgimento da tecnologia de rede local, LAN (do inglês, *Local Area Network*) e o *software* de HMI baseado no PC (Computador pessoal). O surgimento da tecnologia de rede local, possibilitou aos sistemas SCADA a ligação entre si. Os protocolos LAN eram usados com sistemas proprietários, o que impossibilitava a comunicação com sistemas de outros fornecedores.

No final das décadas de 90 e no início dos anos de 2000, tendo como base os sistemas distribuídos, os sistemas de SCADA adotaram uma arquitetura de sistema aberto e protocolos de comunicação “*open-source*”. Com esta arquitetura os sistemas passaram a designar de sistema de SCADA em rede.

No início do ano 2000, com a evolução de várias áreas como a microeletrónica, sistemas de comunicação e com a evolução do processamento computacional começou a emergir a tecnologia de Internet das Coisas, IoT (do inglês, *Internet of Things*) (7).

Atualmente os sistemas de SCADA são fundamentais para as organizações, pois permitem uma grande eficiência dos processos, comunicação de problemas em tempo real e reduzem os tempos de inatividade.

## 2.4.1. Arquitetura dos sistemas de comunicação

No início das redes de computadores, muitos fornecedores trabalhavam de forma separada no desenvolvimento das suas tecnologias de comunicação. Com a preocupação de interoperabilidade entre sistemas, foi criado um padrão internacional para um protocolo comunicação. Como resultado a ISO (do inglês, *International Standard Organization*) criou um modelo de referência o OSI (em inglês, *Open System Interconnection*) (8).

### 2.4.1.1. Modelo OSI

Os protocolos de comunicação seguem um modelo de referência o, OSI, este foi criado pela organização (do inglês, *International Organization for Standardization*), ISO, em 1970 e formalizado em 1983 (9). O modelo tem o principal objetivo de ser *standard*, para os protocolos de comunicação entre os diversos sistemas, garantindo assim uma comunicação *end-to-end*.

Este modelo está dividido em sete camadas que representam uma divisão das tarefas necessárias para implementar uma rede (10).

Tabela 1. Modelo OSI

| Camada |              | Função   |
|--------|--------------|--|
| 7      | Aplicação    | Partilha de recursos, acesso remoto a ficheiros e terminais virtuais                       |
| 6      | Apresentação | Transferência de dados entre a rede e a aplicação inclui encriptação e compressão de dados |
| 5      | Sessão       | Gestão da comunicação entre nós, sincronização e deteção de falhas                         |
| 4      | Transporte   | Controlo da transmissão, reconhecimento, multiplexagem e controlo de congestionamentos     |

|   |                  |   |
|---|------------------|---|
| 3 | Rede             | Encaminhamento entre nos da mesma rede, controlo de tráfego, <i>routing</i> e endereçamento |
| 2 | Ligação de dados | Método de acesso, composição de tramas  |
| 1 | Física           | Transmite e recebe os bits através de meio físico   |

A criação de normas para a comunicações de dados é necessária para gerir o uso e interligação de equipamentos tanto a nível físico, como elétrico e mesmo a nível dos processos e procedimentos manipulação os dados.

#### 2.4.1.2. Norma IEC 60870-5

A Norma IEC 60870-5 foi produzida pela IEC (do inglês, *International Electrotechnical Commission*). Esta tem como principal objetivo criar um padrão para a transmissão de informação para os sistemas de supervisão. Na norma contém a informação detalhada dos equipamentos a aplicar nos sistemas de controlo remoto (11).

Esta é uma norma internacional usada para comunicações de sistemas de controlo e uma solução para a integração de sistemas de gestão e supervisão. A norma é a mais aplicada nos países europeus.

A norma IEC 60870-5 está dividida em 6 seções (11):

- IEC 60870-5-1 – Transmissão de tramas;
- IEC 60870-5-2 – Serviços de transmissão de dados;
- IEC 60870-5-3 – Estrutura genérica dos dados;
- IEC 60870-5-4 – Definição e codificação da informação;
- IEC 60870-5-5 – Aplicação de funções básicas;
- EC 60870-5-6 – Orientações para testes de conformidade para o protocolo IEC 60870 5.

Com o desenvolvimento tecnológico foram criadas mais 4 seções em complemento às 5 anteriormente criadas, são elas (11):

- IEC 60870-5-101 – Protocolo de transmissão, para controlo de sistemas de comunicação básico;

- IEC 60870-5-102 – Protocolo de transmissão orientado para sistemas elétricos de potência;
- IEC 60870-5-103 – Protocolo de transmissão, usado para a interface informativa da proteção dos equipamentos;
- IEC 60870-5-104 – Protocolo de transmissão para acesso à rede.

## 2.4.2. Arquitetura dos sistemas SCADA

Ao longo dos anos os sistemas de SCADA evoluíram em paralelo com a evolução tecnológica. Durante os anos surgiram quatro gerações de sistema de SCADA (12) :

- Primeira Geração – Monolítico;
- Segunda Geração – Distribuída;
- Terceira Geração – *Networked*;
- Quarta Geração – *Internet of Things*.

### 2.4.2.1. Monolithic SCADA Systems

Numa primeira geração, os sistemas de SCADA centraram-se em sistemas *mainframe*. Este sistema é baseado num computador de grandes dimensões, dedicado normalmente ao processamento de um volume grande de informação. Cada sistema SCADA é independente e praticamente sem conectividade com outros sistemas similares, devido ao facto de não existirem redes distribuídas.

Os protocolos de comunicação aplicados nestas redes de SCADA foram desenvolvidos pelos fornecedores de RTU, o que levava a que não existissem diferentes fornecedores dentro das redes. Estes protocolos eram limitados, uma vez que só permitiam a análise, controlo e intercâmbio dos dados entre o computador mestre e os sensores de campo ou os atuadores do terminal remoto.

A redundância nestes sistemas é efetuada com o uso de dois sistemas de *mainframe* equipados de forma idêntica, conectados no nível do barramento (12).

A Figura 2 mostra um exemplo destas redes. Verifica-se que as ligações são feitas diretamente entre o *SCADA Master*, o *mainframe*, e os *RTU's* que estão a comunicar

dentro de uma rede de longa distância. Estas redes de longa distância, WAN (do inglês *Wide Area Networks*), eram projetadas unicamente para a comunicação entre os RTU no campo, pois muitos dos protocolos de redes de longa distância ainda eram desconhecidos (12).

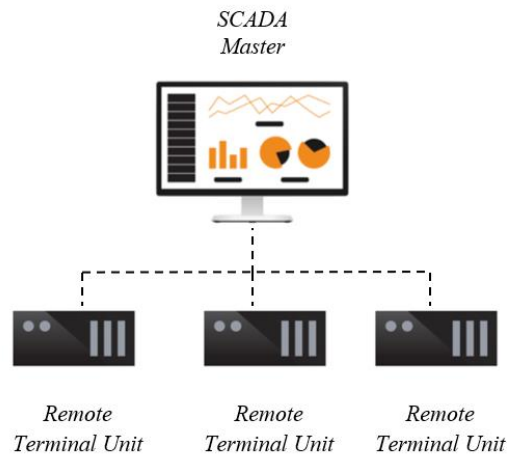


Figura 2. Exemplo de uma rede SCADA: *Monolithic*.

A inexistência de comunicação entre os diferentes RTU's para um computador mestre, levou a que os fabricantes evoluíssem para melhorar essa desvantagem.

#### 2.4.2.2. Distributed SCADA Systems

Nesta segunda geração dos sistemas de SCADA aproveitou-se o desenvolvimento e melhoria da miniaturização dos sistemas e da evolução da tecnologia do LAN. Foram estas as principais razões para os sistemas de SCADA distribuídos.

Várias estações com várias funções foram capazes de comunicar em tempo real, devido ao facto de serem todas ligadas a uma rede LAN. Geralmente estas estações tinham uma unidade de controlo, de menor dimensão e menos dispendiosas do que os equipamentos de primeira geração (12).

Durante esta segunda geração, surgiram os componentes comuns dos sistemas SCADA que são atualmente usados. As estações já podiam conter interfaces homem-máquina, HMI, processadores para a realização de cálculos e ainda base de dados. As estações eram divididas conforme as funções requeridas, fornecendo mais capacidade de processamento

para o sistema. As redes que ligavam esses sistemas individuais eram geralmente baseadas em protocolos de LAN estando confinadas aos limites da rede.

Os protocolos LAN aplicados neste sistema de SCADA eram de origem proprietária, em que o fornecedor criava o seu próprio protocolo de rede. Isso permitia a possibilidade de aumento das velocidades de comunicação, a confiabilidade do sistema e a otimização do tráfego em tempo real. No entanto, limitava a conexão da rede de outros fornecedores.

Mesmo que os sistemas fossem mais confiáveis devido à funcionalidade distribuída, estes estavam limitados ao *hardware*, *software* e dispositivos periféricos proprietários.

Na Figura 3 é possível observar um exemplo desta geração. Podemos verificar que a *Operating Station* está ligada a uma rede LAN, que comunica com um *Communications Server* e com os *RTU's*.

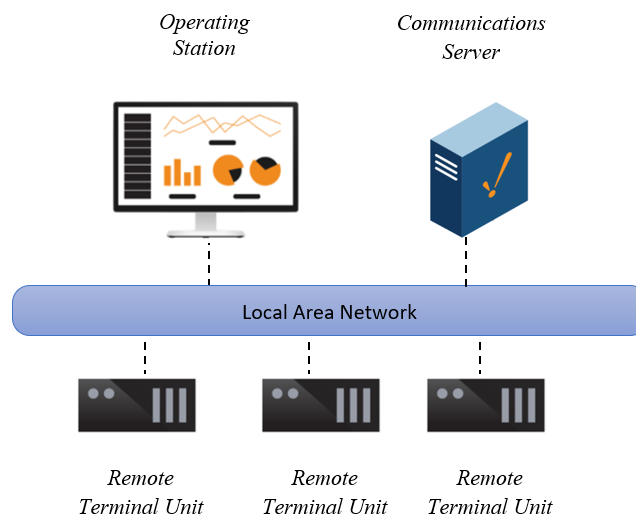


Figura 3. Exemplo de uma rede SCADA: *Distributed*

Este sistema não só serviu para um aumentar a capacidade de processamento, mas também para melhorar a redundância e a confiabilidade do sistema como um todo. Em vez de um esquema de ligação de RTU e do *SCADA Master*, no qual o RTU só era ativo quando se estabelecia a comunicação, neste caso todas as estações da rede estão sempre ativas. Assim se uma estação HMI apresentar uma falha, todo o sistema é informado, permitindo emitir alarmes.

As redes de longa distância ainda são aplicadas para a comunicação entre as estações locais e o mestre do SCADA. Estas redes de comunicação externas ainda estavam muito

limitadas aos protocolos RTU e não estavam disponíveis para outros tipos de tráfego de rede.

Como na primeira geração de sistemas, a segunda geração SCADA também se limitou ao *hardware*, *software* e dispositivos periféricos indicados pelo fornecedor.

### 2.4.2.3. Networked SCADA Systems

O crescimento de toda a indústria, o aumento dos processos automatizados e também a multiplicidade de fornecedores de equipamentos industriais, são as principais causas para a próxima evolução dos sistemas de SCADA.

Esta terceira geração é muito semelhante a segunda geração, no entanto distingue-se por ser orientada para uma arquitetura de sistema aberto, em vez de um ambiente proprietário e controlado pelo fornecedor.

Uma das principais melhorias na terceira geração é a abertura da arquitetura dos sistemas, e a utilização de protocolos abertos. Com esta melhoria elimina-se as principais limitações das anteriores gerações. A utilização destes protocolos tornou mais fácil a ligação de dispositivos periféricos de terceiros, como por exemplo monitores, impressoras entre outros (12).

Um fator-chave que ajudou o rápido desenvolvimento da terceira geração de sistemas SCADA foi o desenvolvimento de novos protocolos para as redes de longa distância, como o protocolo IP, (do inglês, *Internet Protocol*), para comunicação entre a estação mestre e os periféricos. Assim, todos os componentes do SCADA poderiam comunicar uns com os outros através de ligações TCP/IP sobre redes Ethernet (12).

Esta terceira evolução só foi possível devido ao esforço dos fornecedores atender as necessidades das indústrias.

A outra razão foi o desenvolvimento de uma série de normas e especificações para telecomunicações industriais. Em 1996, um conjunto de engenheiros da indústria de automação industrial desenvolveram uma norma chamada OLE, (do inglês, *Object Linking and Embedding*) para controlo de processos. Esta série de normas foi um grande passo para tornar o desenvolvimento de sistemas SCADA muito mais fácil e eficiente.

Na Figura 4 é possível observar um exemplo da terceira geração de sistemas de SCADA. A comunicação do “*SCADA master*” entre os RTUs e a “*Communications Server*” é feita através de uma rede de longa distância baseada no protocolo IP.

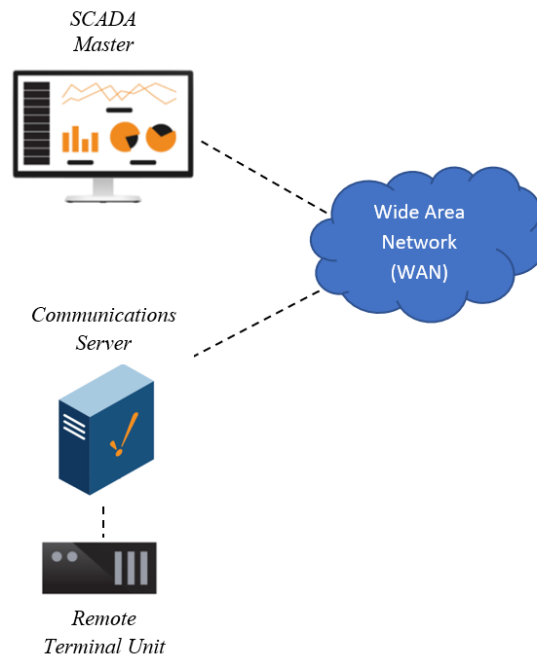


Figura 4. Exemplo de uma rede SCADA: *Networked*

Uma das outras vantagens trazidas por esta distribuição é a robustez a falhas. Isto é, devido ao facto de os sistemas estarem fisicamente separados, torna-se possível construir um sistema SCADA que possa sobreviver a uma perda total de qualquer local. Para algumas organizações que vêm o SCADA como uma função supercrítica, esse é um benefício real.

#### 2.4.2.4. Internet of Things

A quarta geração dos sistemas de SCADA é denominada de internet das coisas, IoT (do inglês, *Internet of Things*), cujo aparecimento foi impulsionado principalmente pelos avanços na computação em nuvem e pelo crescimento contínuo da internet das coisas (13).

A internet das coisas vai complementar o sistema de controlo distribuídos (DCS) e o SCADA, potenciando os seus recursos existentes, como por exemplo a recolha de dados

em tempo real, controlo do processo em tempo real, alarmes de avaria da máquina e anomalia, registos e análise de dados e correspondente visualização (Figura 5).

A aplicação da internet das coisas e as tecnologias de computação em nuvem, como por exemplo *WebHMI* e *HTML5*, habilitou estes sistemas a informar o estado de um processo em tempo real, bem como implementar algoritmos de segurança muito mais complexos (14).

Com o aumento do número de dispositivos ligados, a necessidade de ligar, trocar e analisar informação aumentou substancialmente. As empresas começaram a observar uma necessidade crescente de maior interoperabilidade e transparência de informação para serem competitivas e reduzir os custos.

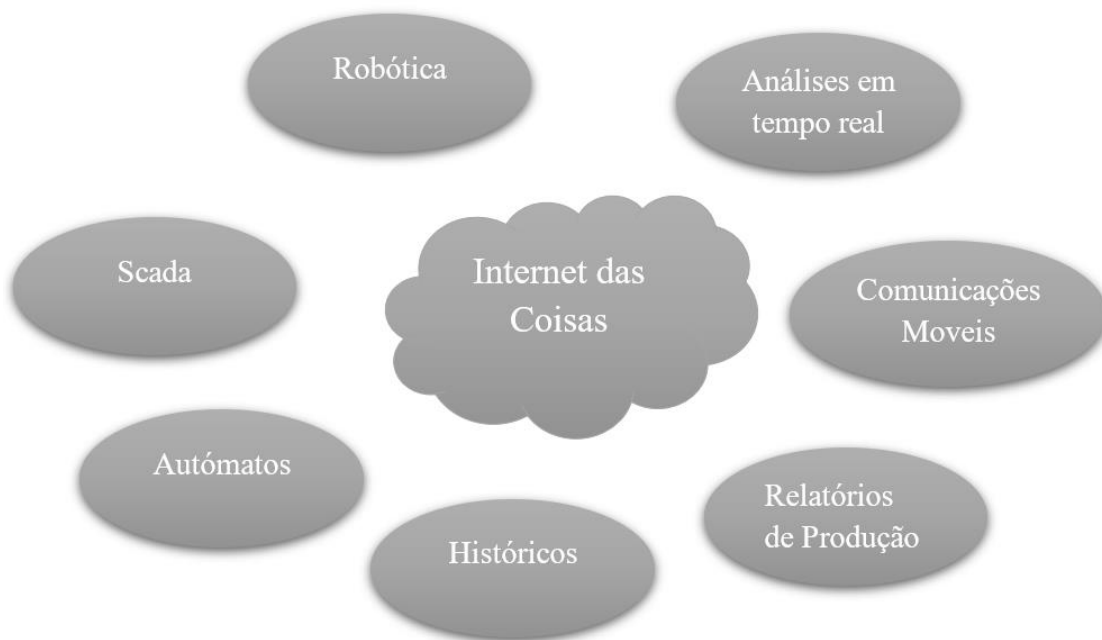


Figura 5. Exemplo de uma implementação do IOT.

Com a implementação e desenvolvimento contínuos de tecnologias como redes de comunicações sem fios 5G, análise avançada de dados baseados em inteligência artificial aprendizagem automática e em segurança cibernética, os sistemas SCADA da quarta geração e os que virão no futuro poderão parecer muito diferentes da primeira geração (15).

A criação dos sistemas de SCADA e IoT têm uma base comum, no entanto possuem algumas diferenças para a sua integração. Os sistemas SCADA são baseados em

protocolos de comunicação aplicados em redes locais e baseado em mestre e escravo. Neste caso, o mestre tem a função de realizar a ligação com o escravo, tendo este que responder. O escravo não comunica com outros escravos e tem uma limitação do número de escravos a ser aplicados numa rede.

Os sistemas de IoT, são baseados em comunicações máquina a máquina (M2M), compostas por um concentrador (*broker*) que medeia a interações entre todos os dispositivos interligados na rede. Os requisitos de rede são mínimos, pois são projetados para dispositivos incorporados com recursos limitados. Contudo tem uma grande eficiência de comunicação (13).

A integração dos sistemas de IoT com os SCADA faz-se aplicando os protocolos existentes para gerir os dispositivos locais. No entanto os protocolos M2M de IoT funcionam como interface para aumentar o alcance dos dados dos dispositivos recolhidos.

### **2.4.3. Protocolos de comunicação**

Os protocolos de comunicação são semelhantes aos idiomas, o que permite que os sistemas de SCADA, as unidades de controlo e as unidades de controlo remoto comuniquem entre eles. Neste relatório são abordados os protocolos Modbus, Ethernet/IP e Profibus.

#### **2.4.3.1. Protocolo Profibus**

O Profibus (do inglês, *Process Field Bus*) é um protocolo de comunicação de origem Alemã. O protocolo é dividido em três variantes Profibus DP, Profibus FMS e Profibus PA.

- Profibus FMS, (do inglês, *Fieldbus Message Specification*) – a primeira a ser criada e é a variante mais universal para tarefas de comunicação complexas. Foi desenvolvida para funções sofisticadas de comunicação entre dispositivos inteligentes, mas com o desenvolvimento das redes Profinet, baseada nos

protocolos Ethernet e UDP (do inglês, *User Datagram Protocol*), tem vindo a ser descontinuada (16).

- Profibus DP, (do inglês, *Decentralized Periphery*) é a mais aplicada e foi criada para operar a uma alta velocidade e ligação de baixo custo. É aplicada no controlo de sistemas de automação e é otimizada para a troca de pequenas mensagens entre dispositivos de entradas e saídas (16).
- Profibus PA, (do inglês, *Process Automation*) esta é desenvolvida para aplicações na indústria de processos. É usada em aplicações intrinsecamente seguras permitindo fornecer alimentação elétrica aos diferentes dispositivos (16).

#### **2.4.3.2. Protocolo Ethernet**

O Ethernet /IP é um protocolo para uma aplicação industrial, mais especificamente aplicações de automação. A criação deste protocolo surgiu de quatro empresas independentes para promover o Ethernet de domínio público para a Automação Industrial. Esses grupos incluem a ODVA (do inglês, *Open DeviceNet Vendor Association*), a IOANA (do inglês, *Associação Industrial Aberta Ethernet*), a CI (do inglês, *Control Net International*) e a IEA (do inglês, *Industrial Ethernet Association*). Neste protocolo é aplicado os protocolos de transporte e controlo aplicados na Ethernet padrão, incluído o protocolo de controlo de transporte (TCP), o IP (do inglês, *Internet Protocol*).

A Ethernet/IP é criado a partir de um padrão amplamente usado em *DeviceNet* e *ControlNet* conhecido como CIP (do inglês, *Control and Information Protocol*). Este padrão organiza dispositivos em rede como uma coleção de objetos. Ele define o acesso, procedimentos e extensões que permitem dispositivos diferentes serem acedidos aplicando um protocolo comum.

O protocolo Ethernet/IP proporciona a aplicação de vários tipos de topologias, desde a estrela convencional com dispositivos de infraestruturas Ethernet padrão, ou por DLR (do inglês, *Device Level Ring*). O DLR tem uma topologia em anel com um só supervisor (17). Este tem a função de gestão do tráfego e de guardar a informação do diagnóstico. Quando existe um problema na rede o supervisor tem de redirecionar o tráfego para

garantir a operabilidade da rede. O tempo de recuperação é de 3ms para uma rede até 50 nós.

A tecnologia de *QuickConnect* permite que os diapositivos sejam trocados enquanto a rede está em funcionamento. Em termos de débitos de rede, estes podem variar desde os 10 Mbps até 1 Gbps (17).

#### **2.4.3.3. Protocolo de Modbus**

O protocolo Modbus foi desenvolvido no ano de 1979, criado pela empresa *Modicon Incorporated*, com o objetivo de integrar autómatos, computadores, terminais, sensores e outros dispositivos de controlo e monitorização.

Este protocolo interage com diferentes tipos de padrões de meios físicos, tais como RS232, RS485 e TCP/IP, assim possibilitando a comunicação através de redes Ethernet. O uso de *gateways* permite a comunicação entre os vários barramentos/redes (18).

Este protocolo comunica com o modelo de um mestre e um ou vários escravos. O mestre pode enviar e escrever mensagens, enquanto que o escravo só pode receber e responder quando pedido pelo mestre. Na mesma linha, cada mestre pode estar ligado a um máximo de 247 escravos (19).

No entanto os meios físicos não suportam mais de 250 escravos. No meio físico RS485, usado neste projeto, estabelece um limite de 32 dispositivos.

O Modbus tem algumas características fixas, entre as quais, o formato das mensagens, a sequência, a verificação dos erros e as funções que executa. Outras podem ser selecionadas pelo utilizador, por exemplo, a escolha do meio de transmissão, o baud rate, a paridade, o número de stop bits e os modos de transmissão (ASCII ou RTU) - estes parâmetros não podem ser alterados com o programa em execução (20).

Neste protocolo o número de escravos (endereços) ligado ao mestre depende do meio físico que é implementado. Por exemplo, usando a ligação RS-232 pode existir um mestre e só um escravo, enquanto na ligação RS-485 pode existir um mestre com vários escravos.

O mestre é responsável por iniciar a comunicação, enquanto o escravo responde aos pedidos. Os escravos nunca transmitem dados a menos que seja solicitado, nem podem comunicar entre eles (20).

Existem dois tipos de mensagens que o mestre pode solicitar ao escravo, *unicast* e *broadcast*.

- *Unicast* – O mestre envia a mensagem a um determinado escravo. Após recebida, a mensagem é tratada e uma resposta é enviada ao mestre (20).

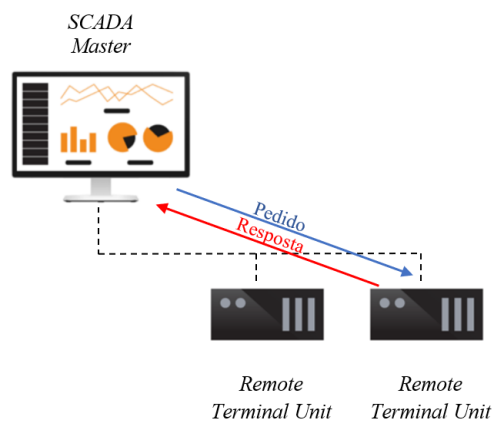


Figura 6. Tipo mensagem: *Unicast*

- *Broadcast* – O mestre envia uma mensagem a todos os escravos. Estes ao receberem a mensagem executam-na, mas não é enviada nenhuma resposta ao mestre (20).

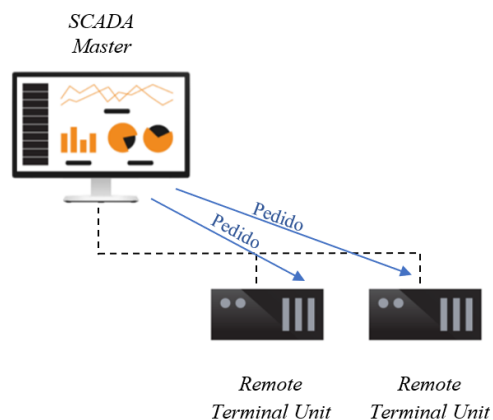


Figura 7. Tipo mensagem: *Broadcast*

### 2.4.3.3.1. Modos de operação do protocolo Modbus

Os modos de transmissão para a comunicação série usando o protocolo Modbus pode seguir dois modos:

- Modbus RTU – Este modo é utilizado quando a informação é transmitida em formato binário;
- Modbus ASCII – Este modo é aplicado na codificação da mensagem em ASCII (do inglês, American Standard Code for Information Interchange)

Não é permitido usar os dois modos de transmissão na mesma rede.

Na comunicação Modbus em RTU, quando em linha série, cada byte (8 bits) da mensagem é enviado através de 2 caracteres hexadecimais, isto é, uma “palavra” (“*word*”). Cada mensagem deve ser transmitida uma sequência de “*words*”. A principal vantagem deste modo de transmissão é o facto da sua grande densidade de caracteres permitir um processamento de dados mais eficaz do que o modo ASCII (para o mesmo *baud rate*) permitindo um aumento do desempenho da comunicação. A verificação dos erros é baseada no algoritmo de CRC-16 (20). Na Figura 8 é possível observar a estrutura *frame* do Modbus do modo RTU.

| Protocol Data Unit (PDU)    |               |               |                            |           |             |
|-----------------------------|---------------|---------------|----------------------------|-----------|-------------|
| Start                       | Slave Address | Function code | Data                       | CRC Check | End         |
| >= 3,5 Char                 | 1 Byte        | 1 Byte        | N * Bytes<br>(0-252 Bytes) | 2 Bytes   | >= 3,5 Char |
| Application Data Unit (ADU) |               |               |                            |           |             |

Figura 8. Estrutura de um *frame* Modbus em RTU

No modo RTU, o quadro de mensagem inicia com um intervalo silencioso de pelo menos 3,5 caracteres e termina com um intervalo silencioso de pelo menos 3,5 caracteres.

A mensagem deve ser transmitida por um fluxo constante de caracteres. Se um tempo de silêncio maior do que 1,5 tempos de caracter (definido internamente) for detetado, o quadro da mensagem é declarado incompleto e deve ser descartado pelo recetor (18).

No modo de transmissão ASCII, cada byte de uma mensagem é enviado como dois caracteres ASCII, deste modo é menos eficiente que o modo RTU, uma vez que para cada byte são precisos dois caracteres. A verificação de erros é feita pelo algoritmo LRC, (do inglês, *Longitudinal Redundancy Check*) (20). Na Figura 9, temos uma estrutura de um *frame* Modbus em modo de ASCII.

| Start  | Slave Address | Protocol Data Unit (PDU) |                         | LRC Check | End             |
|--------|---------------|--------------------------|-------------------------|-----------|-----------------|
|        |               | Function code            | Data                    |           |                 |
| 1 Char | 2 Chars       | 2 Chars                  | N char (s)<br>(0-2*252) | 2 Chars   | 2 Chars<br>CRLF |

Figura 9. Estrutura de uma *frame* Modbus em ASCII

Assim que o mestre envia uma mensagem, coloca-a num quadro, que define inequivocamente o início e o fim da mesma. Isto permite que os dispositivos recetores que recebam a mensagem conheçam o início e fim desta. Ao ser detetadas mensagens parciais é gerado um código de erro que é enviado para o *master* como resposta.

#### 2.4.4. Componentes do Sistema SCADA

Na Figura 10, estão descritos os componentes mais importantes de um sistema de SCADA. Os sensores ou rede de sensores são responsáveis por detetar sinais físicos que constituem a informação que é tratada pelo sistema de controlo. Por sua vez, os RTU's ou os autómatos são responsáveis pela recolha e tratamento dos dados vindos dos sensores e o controlo de todo o processo. Por fim, os HMI ou interfaces SCADA realizam a interface homem-máquina.

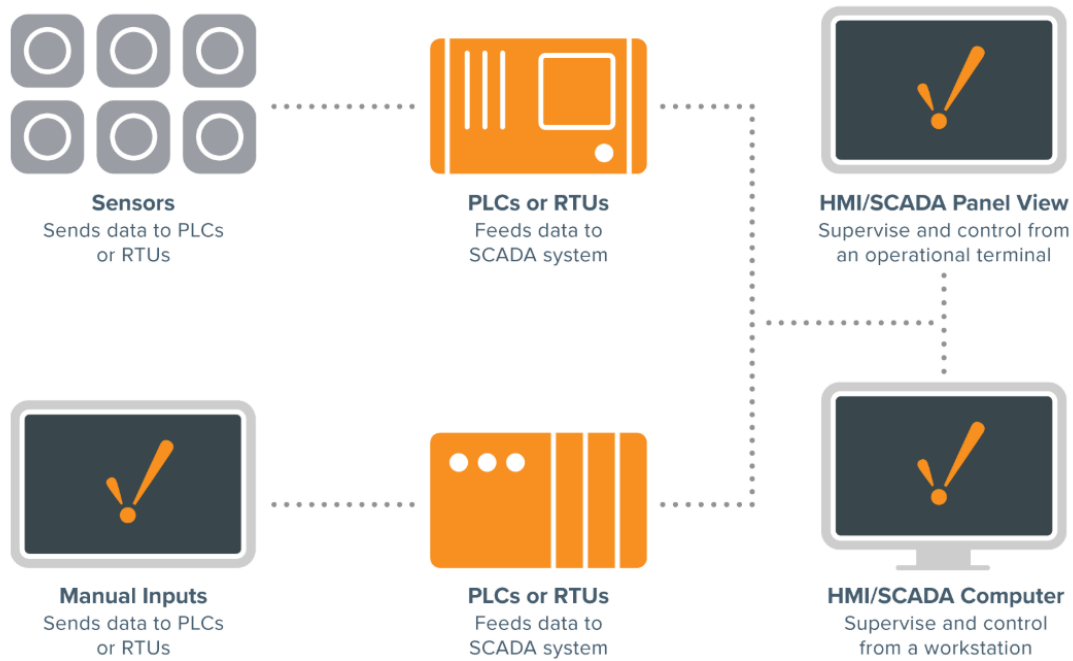


Figura 10. Rede de um Sistema de SCADA (6)

#### 2.4.4.1. Unidades de Terminal Remoto (RTU)

Esta unidade tem como principal função a recolha de dados de sensores que estão no campo e convertê-los em dados digitais. Estes são enviados para os autómatos ou para os sistemas de supervisão por meios de comunicação.

#### 2.4.4.2. Controlador lógico programável (PLC)

As funções desta unidade são muito semelhante às unidades de (RTU). Assim, esta pode controlar localmente o processo e executar lógicas simples e complexas. Esta reúne as informações colhidas pelos sensores, atuadores e pelo RTU e é capaz de trocar dados com o controle de supervisão.

Esta unidade pode-se ligar a um módulo de HMI diretamente. Sendo assim, o PLC pode ser controlado localmente ou remotamente.

#### **2.4.4.3. Interface Homem-Máquina (HMI)**

O HMI (Interface Homem-Máquina) é um componente que apresenta os dados recolhidos pelos dispositivos locais, permitindo que o operador interaja com o processo.

O HMI tem uma grande importância nos sistemas de SCADA pois através destes os operadores podem interagir com o processo, fazendo diagnósticos, verificar os alarmes do processo, gerar relatórios de produção e verificação de gráficos de produção.

#### **2.4.4.4. Base de dados**

As bases de dados, num sistema de SCADA, servem para registar o histórico de produção e para suporte à gestão da produção. Estas podem registar a hora e data, eventos, alarmes ou qualquer outra informação recolhida pelo sistema. A base de dados pode ser acedida a partir do HMI para a gestão de relatórios de produção.

#### **2.4.4.5. Supervisão**

Esta é a unidade mais importante num sistema de SCADA é geralmente um computador ou um *PainelPC*, que contém um *software* específico onde se pode controlar os diferentes dispositivos associados ao sistema, podendo visualizar o processo em tempo real.

### **2.4.5. Software SCADA: Movicon**

O *software* Movicon é desenvolvido pela empresa Progea, neste projeto aplicou-se a versão 11. Esta é específica para monitorização e controlo de processos.

O Movicon 11 é a solução para todos os processos que precisam do desenvolvimento de aplicações de *software* de supervisão, controlo e aquisição de dados com interface de operação, onde os dados podem ser acedidos remotamente ou localmente. Esta é uma plataforma de SCADA que garante flexibilidade de desenvolvimento, versatilidade e independência (21).

Esta plataforma é muito versátil e com muitos modos de operação, devido a ser totalmente baseada em XML (do inglês, *Extensive Markup Language*) uma metalinguagem definida pelo padrão internacional do consórcio W3C. O XML permite que todos os arquivos sejam abertos e editados com qualquer outro editor. Os projetos podem ser agrupados em estruturas hierárquicas, distribuídos e compartilhados.

O Movicon possibilita a escrita e desenvolvimento de scripts em várias linguagens, tais como VBA (Visual Basic para Aplicações) e Visual Basic.Net (21).

Em termos de base de dados estas podem ser guardadas em qualquer *software* dedicados a estas. Os registos da base de dados são geridos através do drive ODBC, (do inglês, *interface open database connectivity*). Este drive usa a ODBC da Microsoft, permitindo que as aplicações tenham acesso aos sistemas de gestão da base de dados DBMS, (do inglês, *database management system*), usando a linguagem de programação SQL, (do inglês, *Structured Query Language*) como padrão para gerir os dados (22).

Além das características já mencionadas, esta plataforma inclui ferramentas de análise e desenvolvimento de relatórios, independentemente de onde são guardados de onde são guardados os dados. Esta contém também vários objetos para a análise de dados em tempo real como tendências, gráficos, tabelas. O *Report Designer* permite criar e gerir os relatórios localmente (21).

## 2.4.6. Comunicação com a base de dados

A comunicação com a base de dados é feita através do ADO, (do inglês, *Microsoft ActiveX Data Objects*), este permite que os aplicativos clientes acessem e manipulem dados de várias fontes por meio de um fornecedor de OLE DB para ODBC.

O ADO suporta os principais recursos para a criação de uma aplicação de cliente / servidor baseados na Web (23).

A arquitetura geral da integração de ADO's num sistema é ilustrada Figura 11.

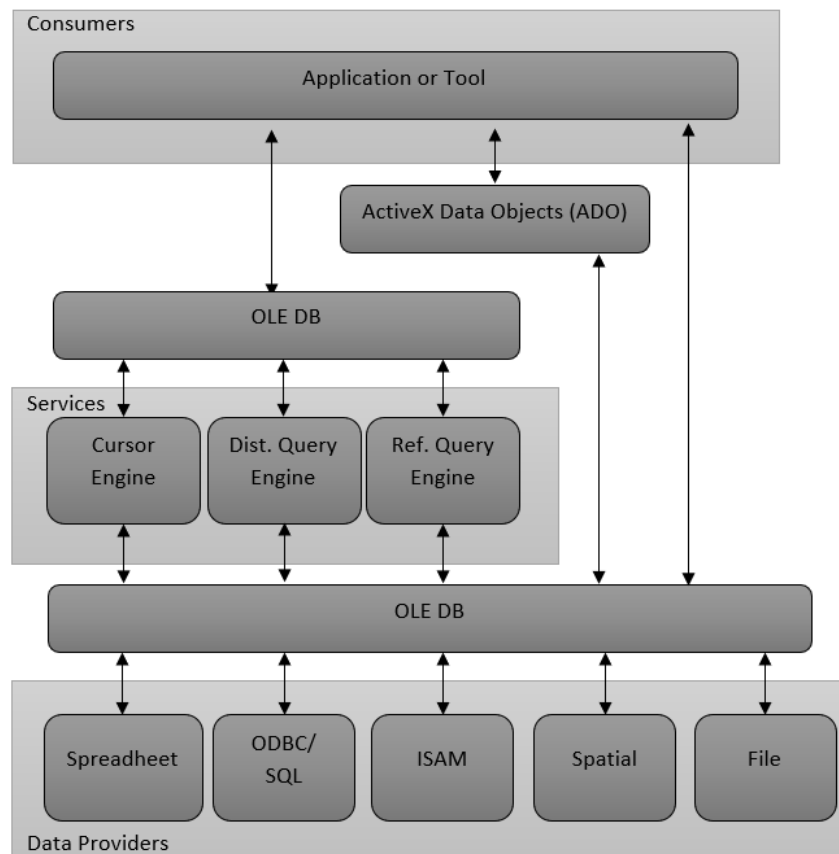


Figura 11. Arquitetura geral da integração de ADO's

### **3. Processo de Rotomoldagem**

Este capítulo apresenta o processo de rotomoldagem, onde é explicado os tipos de materiais aplicados no processo e tipos de máquinas desenvolvidas para o mesmo.

### 3.1. Rotomoldagem

A rotomoldagem rotacional, também conhecida como moldagem rotacional é um processo mecânico onde são moldadas peças de plástico utilizando matéria-prima não sólida. Embora exista outros tipos de métodos, como a moldagem por sopro, a termoformação e moldagem por injeção, a rotomoldagem é a única viável para a produção de objetos de grandes dimensões ( $> 2 m^3$ ) em peças únicas. Este método é conhecido pela fabricação de tanques, mas também é utilizada em brinquedos e embarcações. Na Figura 12 e na Figura 13, são apresentados dois exemplos.



Figura 12. Reservatório Vertical em Polietileno



Figura 13. Reservatório subterrâneo em polietileno

O processo de rotomoldagem é um método simples. Consiste em introduzir uma quantidade específica de polímeros em granulado num molde. Esse molde é rodado em torno de dois eixos principais em velocidades relativamente baixas à medida que é aquecido de modo a que o plástico contido no molde adira e forme uma camada monolítica contra a superfície do molde, através da força centrífuga. A rotação do molde continua durante a fase de arrefecimento até que o plástico retenha a forma desejada à medida que se solidifica.

Na Figura 14 podem-se verificar as etapas do método de rotomoldagem. Na alínea (a) observamos o carregamento do molde, na etapa (b) verificamos o aquecimento e a rotação dos eixos, na (c) o arrefecimento do molde com a rotação dos eixos e em seguida temos a desmoldagem do molde (d) (24).

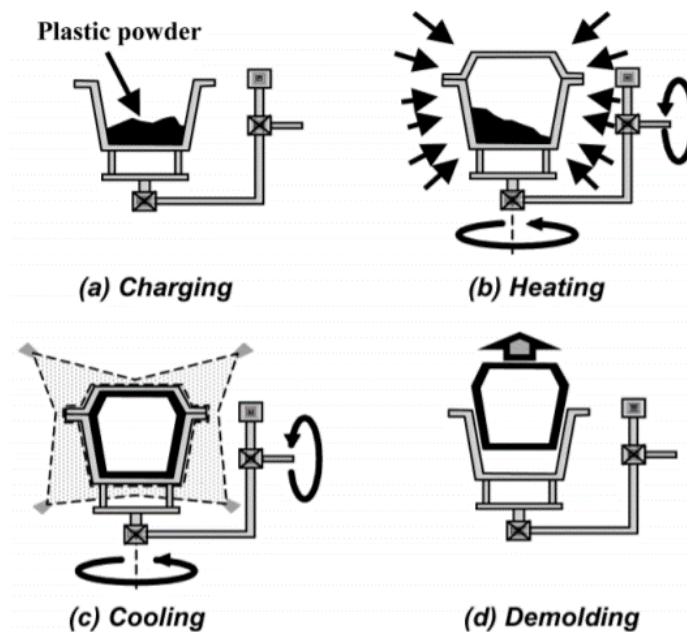


Figura 14. Processo de rotomoldagem (24)

Este método apresenta algumas vantagens tais como: as peças não ficam com linhas de solda ou juntas; os moldes são relativamente mais baratos; fabricação mais rápida do que os outros tipos de máquinas; não existe desperdício de materiais; diferentes tipos de produtos podem ser moldados juntos e assim é possível fazer produtos multicamadas.

No entanto, as desvantagens deste método são: os tempos de fabricação são longos; a escolha de materiais de moldagem é limitada e algumas características geométricas são difíceis de moldar (24).

## **3.2. Tipos de materiais aplicados na rotomoldagem**

De todo os plásticos usados no mundo inteiro, cerca de dois milhões de toneladas, 80% são termoplásticos e 20% são termoendurecíveis. Os polímeros termoendurecíveis podem sofrer alterações químicas durante o processamento, de tal forma que a sua estrutura molecular final é tridimensional. Estes polímeros podem ser comparados a um ovo cozido, pois no final da cozedura não é possível amolecê-lo com um novo reaquecimento. A estrutura molecular final dos polímeros termoendurecíveis é tão complexa que não podem ser reutilizados ou reciclados com os meios convencionais.

Em relação aos polímeros termoplásticos, a estrutura molecular final, após serem processados é essencialmente a mesma que a estrutura molecular original. Estes polímeros podem ser comparados com esparguete. Quando a massa está fria os fios ficam imóveis, mas quando está quente tornam-se flexíveis. A massa também pode ser aquecida e reaquecida repetidamente, sem se perderem as características.

Os Polietileno, polipropileno, poliestireno e policloreto de vinila são os polímeros termoplásticos mais comuns e mais frequentemente. Estes polímeros são considerados reutilizáveis e recicláveis se durante o processamento a estrutura não for danificada mecanicamente (24).

O polietileno é o termoplástico mais aplicados na indústria de moldagem rotacional.

### **3.2.1. Polietileno de Baixa Densidade (LDPE)**

O polietileno de baixa densidade tem uma faixa de densidades de  $0,910 \text{ g / cm}^3$  a  $0,940 \text{ g / cm}^3$ . Em termos químicos este tem um alto grau de ramificação de cadeias curtas e longas. Significa então que as cadeias moleculares não se encaixam na estrutura cristalina, o que resulta numa menor resistência a quebra e num aumento do grau de deformação que um material suporta até ao momento da sua fratura. O índice de fusão é compreendido entre  $100 \text{ °C}$  a  $120 \text{ °C}$ .

Este material é caracterizado pela sua resistência alta, pela resistência química, baixa absorção de água, baixo custo e facilidade de moldar pelos diferentes métodos (24).

### 3.2.2. Polietileno de Alta Densidade (HDPE)

O Polietileno de alta densidade, tem uma densidade maior ou igual a  $0,941 \text{ g/cm}^3$ . Este tipo de polietileno possui um baixo grau de ramificação, o que significa que possui forças intermoleculares mais resistentes à tração. Devido à falta de ramificação, é necessário o uso de um catalisador, como por exemplo de crómio / sílica, *Ziegler-Natta* ou de metaloceno. O índice da temperatura de fusão é a partir dos  $190 \text{ }^\circ\text{C}$ .

Este material tem como características a sua flexibilidade, a translucidez/ ceroso, a boa tenacidade a baixa temperatura (até  $-60 \text{ }^\circ\text{C}$ ), baixo custo e uma boa resistência química (24).

### 3.2.3. Polietileno de Média Densidade (MDPE)

O Polietileno de média densidade, tem uma densidade de  $0,926 \text{ g/cm}^3$  até  $0,940 \text{ g/cm}^3$ . O MDPE é caracterizado por possuir menos cadeias laterais e mais curtas que o LDPE. Devido a este facto, é preferível a utilização de MDPE em relação ao uso de LDPE, devido à sua maior rigidez e facilidade de processo. Em termos de moldagem, as temperaturas necessárias nos MDPE são mais altas que nos LDPE's. Adicionalmente, densifica completamente, têm menos manchas superficiais e menor porosidade do que HDPE's. Necessitam de um catalisador de crómio/sílica, catalisadores de *Ziegler-Natta* ou catalisadores de metaloceno (24).

### 3.2.4. Polietileno linear de baixa densidade (LLDPE)

O polietileno linear de baixa densidade, é comparado na figura em baixo com o HDPE e LDPE. As cadeias são drasticamente reduzidas em comprimento, enquanto que as cadeias são similares às do LDPE. A faixa de densidade situa-se entre  $0,910 \text{ g/cm}^3$  até  $0,940 \text{ g/cm}^3$ . Possui uma melhor rigidez, a resistência química e a resistência à tração, no entanto apresenta uma resistência ao impacto um pouco menor que no LDPE e no MDPE. Este material é procurado pela sua excelente resistência a altas temperaturas de cerca de  $100 \text{ }^\circ\text{C}$  (24).

Na Figura 15 é possível observar as diferentes ramificações dos polímeros, entre os quais HDPE, LDPE, LLDE.

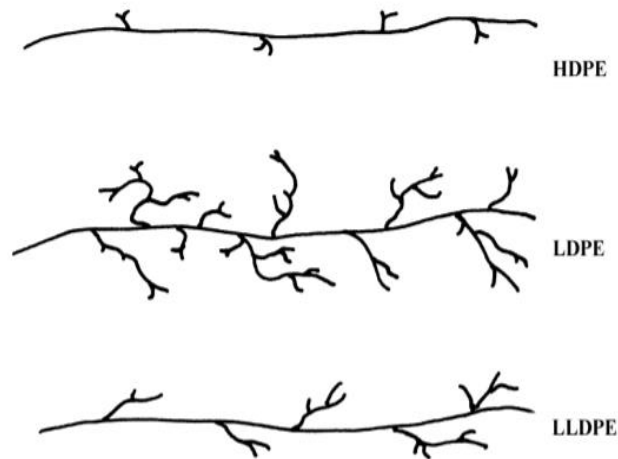


Figura 15. Exemplo da ramificação dos polímeros (24)

### 3.2.5. Etileno Vinil Acetato (EVA)

O Etileno Vinil Acetato é um copolímero de etileno e acetato de vinila. Este polímero é constituído por uma gama de 10% a 40% de acetato de vinila, sendo a restante composição do polímero etileno. O aumento da concentração de acetato de vinila resulta na diminuição da cristalinidade, aumentando a ductilidade e diminuindo a resistência à tração. As densidades típicas de EVA são de  $0,930 \text{ g/cm}^3$  a  $0,950 \text{ g/cm}^3$ . As temperaturas de fusão de EVA variam entre  $90 \text{ }^\circ\text{C}$  a  $120 \text{ }^\circ\text{C}$  diminuindo com o aumento do teor de acetato de vinila (24).

### 3.3. Tipos de máquinas de moldagem rotacional

#### 3.3.1. Rock-and-Roll

Este tipo de máquina, foi uma das primeiras utilizadas na rotomoldagem. A ação de balanço sobre um eixo ("rock") e uma rotação total de 360 graus em torno de um eixo perpendicular ("roll"). Durante muito tempo, esta máquina fazia produtos que eram aproximadamente simétricos em relação a um eixo central, como postes de iluminação, canoas e caiaques (24).

O funcionamento destas máquinas inicia-se ao ser colocado um único molde a uma velocidade de rotação baixa e com ângulo de oscilação menor que 45 graus. O aquecimento do molde é efetuado através de um queimador, sendo que a proximidade do queimador ao molde é um fator importante no aquecimento do molde, conforme se pode observar na Figura 16.

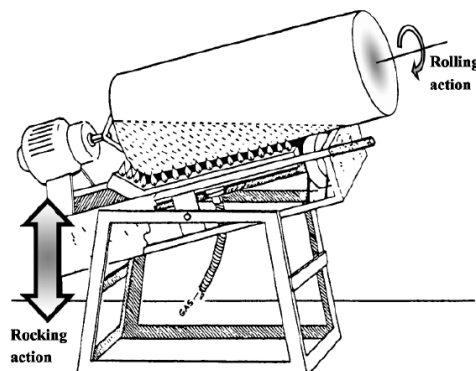


Figura 16. Máquina de *Rock-and-roll*

### 3.3.2. Clamshell

A Figura 17 mostra um exemplo da máquina de *Clamshell*. Esta máquina é constituída por uma tampa horizontal para fechar o forno e os moldes são montados no eixo principal. A máquina fornece rotação biaxial, nos dois eixos, completa e tem a vantagem de o eixo horizontal poder ser suportado em ambas as extremidades (24).

O ar aquecido circula até a temperatura apropriada do polímero ser alcançada. O arrefecimento ocorre por ar arrefecido.

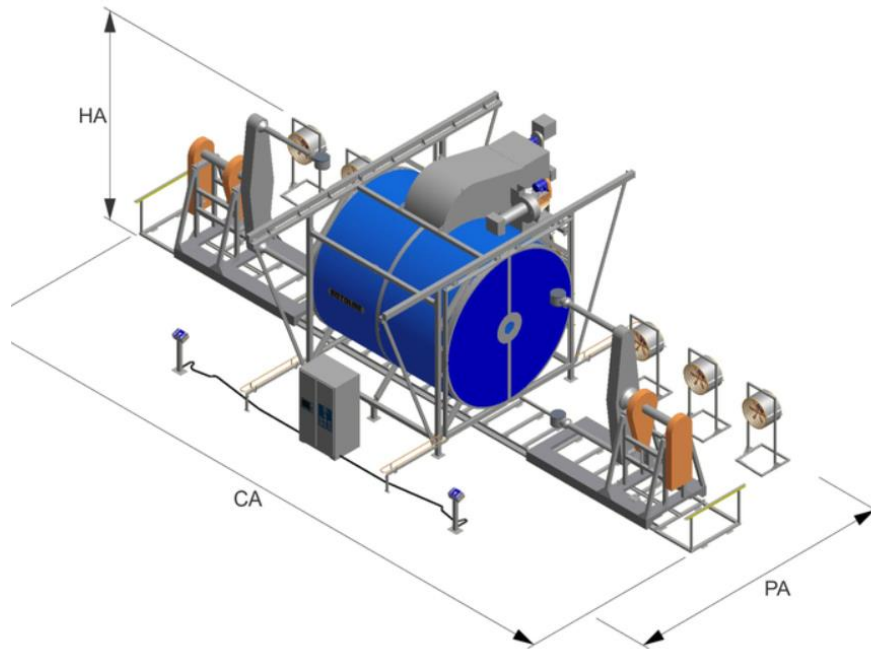
Quando o ciclo de arrefecimento é concluído, a porta do forno abre, o molde é aberto e as peças são retiradas. Os moldes serão limpos, inspecionados e recarregados para o próximo ciclo.



Figura 17. Máquina de *Clamshell*

### 3.3.3. Shuttle Machines

A Figura 18, mostra um esquema da máquina *Shuttle*. A máquina foi desenvolvida para fabricar eficientemente uma ampla variedade de peças que sofrem o processo de rotomoldagem. A *Shuttle* trabalha um conjunto de dois carros, um forno central e duas estações de arrefecimento. É deste tipo a máquina desenvolvida no âmbito deste projeto.

Figura 18. Máquina *Shuttle* (25)

### 3.3.4. Fixed-Arm Carousel Machine

A Figura 19 mostra um esquema de uma máquina *Fixed-Arm Carousel*. A *Fixed-Arm Carousel*, também conhecida por carrossel, torre ou máquina rotativa foi desenvolvida para longas produções e peças até 7 metros e para produzir tanques até 50.000 litros. Neste momento estas são as máquinas mais comuns na indústria. As máquinas mais antigas, têm três braços que estão desfasados 120 graus, movidas por um motor central. Cada braço passa por uma estação de aquecimento, arrefecimento e serviço (24).

Uma das maiores vantagens deste processo de moldagem rotacional é que se podem usar diferentes moldes e diferentes materiais em cada braço. É, assim, possível alterar as combinações de moldes em cada braço em intervalos regulares para que haja grande versatilidade nos cronogramas de produção.

Uma desvantagem das máquinas de braço fixo é que, para um uso otimizado, os tempos de aquecimento, arrefecimento e serviço precisam ser concertados.

Atualmente estas máquinas tem quatro braços, com os braços afastados 90 graus. Geralmente este quarto braço tem uma estação auxiliar de arrefecimento, enquanto os outros braços estão nas outras estações.

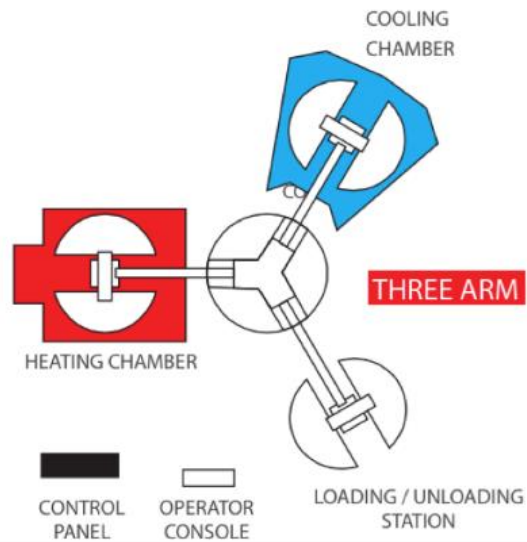


Figura 19. Máquina de carrossel

### 3.3.5. Independent-Arm Machine

Recentemente foram desenvolvidas máquinas de braços independentes para poder melhorar a versatilidade das máquinas rotativas. As máquinas atuais têm cinco estações designadas e podem ter dois, três ou quatro braços que se encaixam independentemente um do outro. Este sistema permite assim ao operador gerir as estações vazias com estações de forno auxiliares e estações de resfriamento auxiliares para separar as etapas de carga e descarga nas estações de serviço (24).

## **4. Implementação da máquina de Rotomoldagem**

Neste capítulo é apresentado o projeto e implementação da máquina de rotomoldagem desenvolvido no âmbito do estágio. É descrito o funcionamento da máquina no processo de rotomoldagem, passando por todos os equipamentos usados para a implementação assim como o *software* de controlo do processo e do sistema de supervisão.

## 4.1. Funcionamento da máquina

A Figura 20 apresenta um esquema da máquina de rotomoldagem. A máquina desenvolvida para a execução do processo de rotomoldagem foi o modelo *Shuttle Machines*, constituída por dois carros e uma estufa.

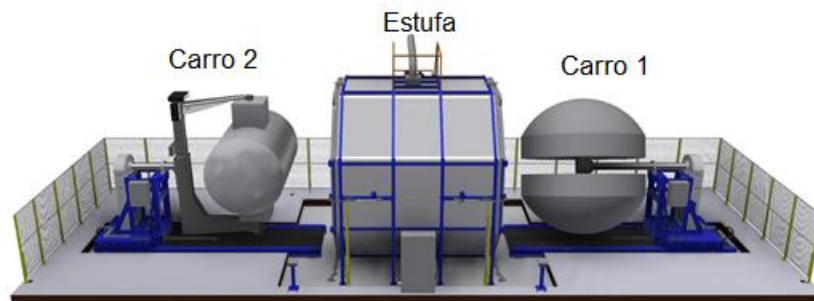


Figura 20. Máquina de Rotomoldagem

A estufa tem como missão o aquecimento do polímero em granulado, para que este se funda e fique no estado líquido de modo a distribuir-se de maneira homogénea pelo molde.

A gestão da estufa é feita através do SCADA e é dividida em cinco patamares que permitem o aquecimento e arrefecimento da estufa. Em cada uma pode-se determinar a temperatura, duração do patamar, velocidade eixo 1 e eixo 2, permitindo assim a gestão do aquecimento do forno.



Figura 21. Estufa

Os carros têm como principal função a rotação do molde em 360 graus, para uma mistura uniforme dos polímeros.

O primeiro carro (Figura 22) permite colocar vários moldes para criar pequenas peças de polímeros e o segundo carro tem só um molde para fazer tanques com capacidade de 4000 litros. Estes carros são modulares e permitem que se configurem os moldes usados. Os carros vão deslocar o molde, através de um carril, até este estar dentro da estufa ou na seção de arrefecimento. O controlo de velocidade é feito através de dois fins de curso que estão desfasados entre si, para ter uma velocidade mais reduzida no final do carril.



Figura 22. Carro 1

A Figura 23, apresenta o esquema dos equipamentos de controlo aplicados neste processo.

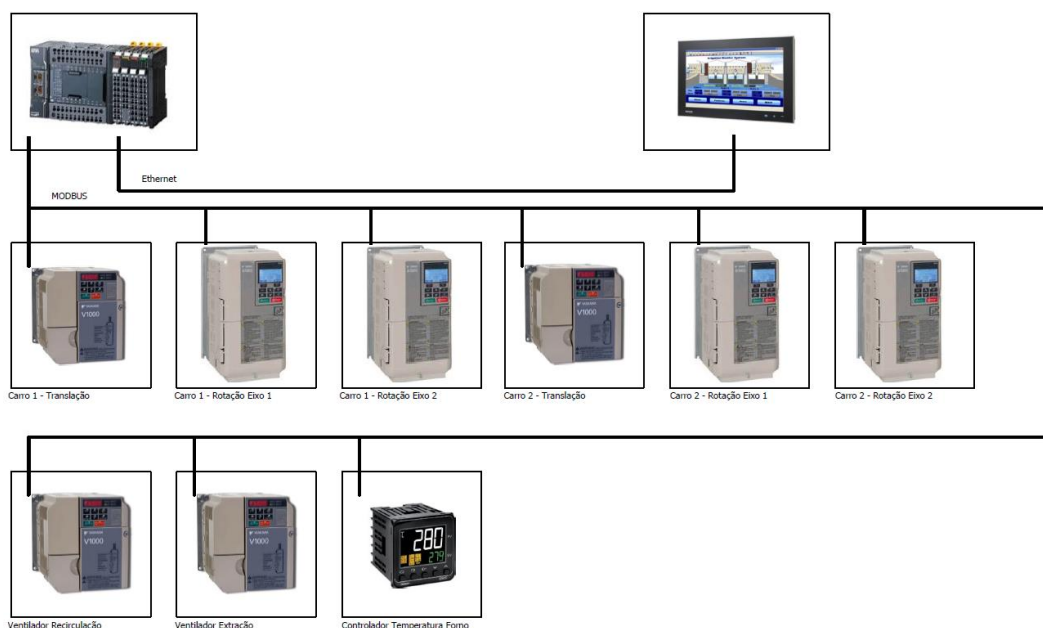


Figura 23. Componentes de controlo da máquina.

## 4.2. Variadores de velocidade

A variação da velocidade dos motores de indução é determinada pela frequência da tensão de alimentação, pelo seu número de polos e pela carga. Por estes motivos para controlar a velocidade do motor, sem recurso a dispositivos mecânicos externos, é necessário a variação da sua frequência. Essa variação de frequência é feita pelos Variadores Eletrônicos de Velocidade.

Os Variadores Eletrônicos de Velocidade são dispositivos de eletrônica de potência que substituem com enormes vantagens todos os sistemas até agora utilizados para o arranque dos motores de indução. Estes têm uma panóplia de variáveis que podem ser ajustadas nomeadamente o binário de arranque, tempo de arranque, tempo de paragem, velocidade da rotação do motor consoante a necessidade pretendida pela aplicação.

Existem vários métodos de variação de velocidade, sendo que os variadores de velocidade são os que proporcionam um maior rigor e precisão. Este são os mais aplicados para o controlo de motores de indução.

Os variadores aplicados no processo são o modelo V1000 e A1000 da marca *Yaskama*.

O V1000 é aplicado nos motores de translação dos carros, nos ventiladores de extração e recirculação.

A Figura 24, apresenta o esquema de ligação aplicado neste projeto. A alimentação dos variadores possui um Disjuntor Termomagnético de três polos para uma corrente nominal de 10 A.

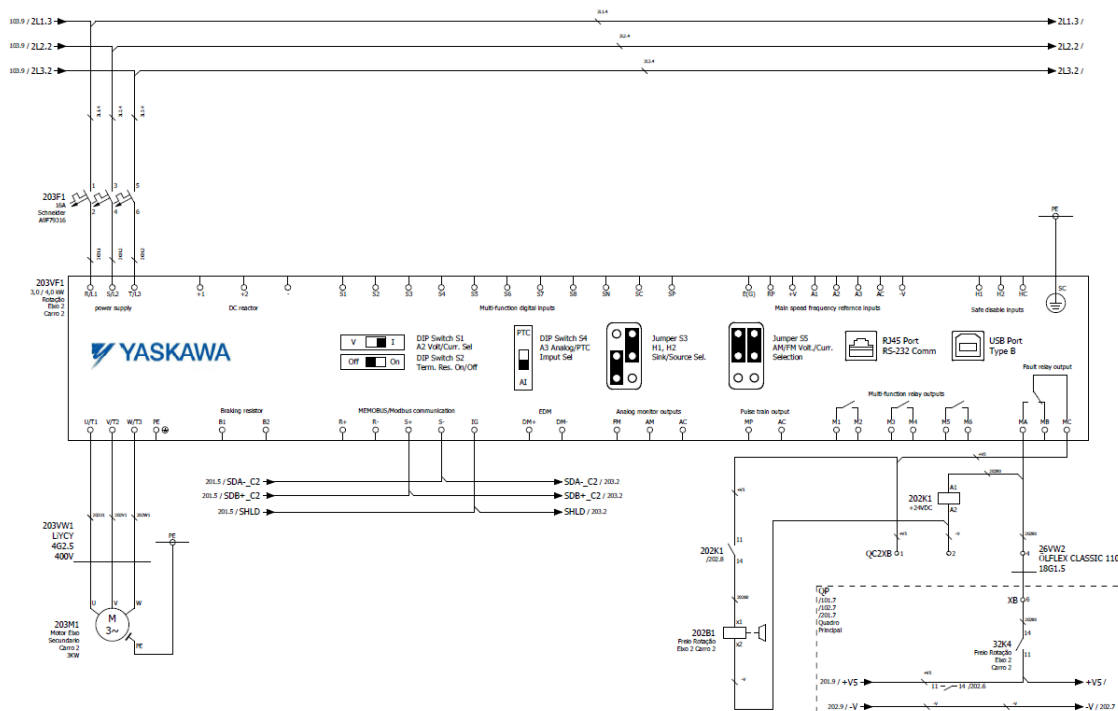


Figura 24. Esquema de ligação do variador A1000.

### 4.3. Minicomputador: Advantech UNO-2362G

O UNO-2362G (Figura 25) da empresa Advantech é um minicomputador de grande performance devido ao seu CPU Dual Core a 1.0GHz e à sua memória RAM de 2GB. Em termos de comunicações este tem dois interfaces Ethernet, porta HDMI, porta RS-232 e RS-485. Este minicomputador é responsável por correr o sistema de SCADA deste processo.



Figura 25. Advantech UNO-2362G

Este computador é próprio para ambientes industriais permitindo o funcionamento em temperaturas desde os  $-20\text{ }^{\circ}\text{C}$  até os  $60\text{ }^{\circ}\text{C}$  e tem uma proteção de vibração até os 5 Grms e contra o choque até aos 50G.

#### 4.4. LCD: FPM-5171G

O FPM – 5171G (Figura 26) é um LCD com um tamanho de 17 polegadas (Figura 26). Este visor de TFT LCD com retroiluminação LED é projetado para aplicações industriais. Em termos de comunicações a transmissão de sinais é feita pelos terminais Direct-VGA e DVI-D, que permitem o uso de placas de controlo VGA, tem uma ficha USB que permite o controlo do teclado por toque. Usa uma faixa de alimentação de 10 a 30 VDC.

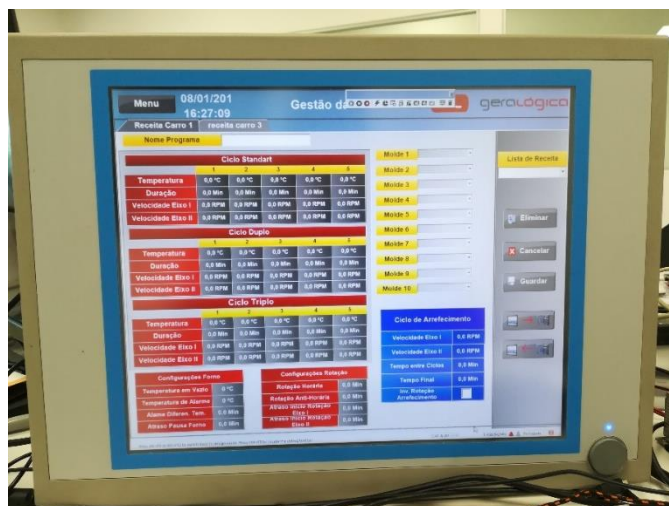


Figura 26. FPM-5171G

## 4.5. Autómatos: NX1P2

O autómato programável utilizado neste processo é o NX1P2, modelo 9024DT1 da Omron (Figura 27).

A aplicação da gama NX1P2 completa a família de controladores de máquinas NX/NJ, oferecendo a mesma funcionalidade num formato compacto.



Figura 27. Autómatos: NX1P2.

O NX1P permite o controlo personalizado de todos os dispositivos das máquinas, tais como movimento, segurança e visão num Ambiente Integrado de Desenvolvimento.

As principais valias destes autómatos em relação a outros são o tempo de ciclo ser rápido (cerca de 2ms) e possuir funções de controlo de movimento até oito eixos, sendo quatro deles sincronizados. Em termos de conectividade possui Portas EtherCAT e EtherNet/IP incorporadas até dezasseis escravos EtherCAT. Podem ainda ser ligadas até duas unidades opcionais para adicionar funcionalidades de comunicação série ou E/S analógicas.

Na Tabela 2, encontram-se descritas as características técnicas mais importantes deste autómato.

Tabela 2. Característica do NX1P2.

| Item                                  |   | NX1P2-9024DT1   |
|---------------------------------------|---|---|
| Fonte de alimentação da unidade       | Alimentação                                 | 24 V  |
|                                       | Corrente consumida                          | Para partida a frio à temperatura ambiente:<br>10 A máx.<br>e<br>2,5 A máx. / 150 ms max<br>(Depende do modo de operação) |
| Comunicações série (Modulo adicional) | Método de comunicação                       | half duplex   |
|                                       | Sincronização                               | Start-stop  |
|                                       | Protocolo suportado                         | Host link, Modbus-RTU master, and no-protocol   |
| Programação                           | Tamanho                                     | 1.5 MB  |
| Capacidade de memória para variáveis  | Retido durante a interrupção de energia     | 32 kB   |
|                                       | Não retido durante a interrupção de energia | 2 MB  |
| Entradas                              | Número de Pontos                            | 14  |
| Saídas                                | Número de Pontos                            | 10  |

Neste projeto foram adicionados componentes da família do NX, que são:

- NX-PF0730- Unidade de fonte de alimentação com as seguintes características:
  - Tensão de alimentação: 5 a 24 VDC
  - Corrente máxima de alimentação de energia de E / S: 10 A.
- NX- ID5442- Módulo de entradas e digitais com 16 pontos. Neste projeto são aplicadas 4 unidades destes módulos, logo possuem 78 entradas digitais neste projeto.
- NX-OD5256- Módulo de saída digital de 16 pontos.

- NX1W-C1F11- Este é um módulo de comunicação com o protocolo de Modbus-RTU, com uma porta RS485, tem uma distância de transmissão de 50m. As comunicações entre os variadores e a sonda de temperatura é feito por este módulo de comunicação usando o padrão de meio físico RS 485. A Figura 28, descreve a ligação deste módulo com a sonda de temperatura do forno.

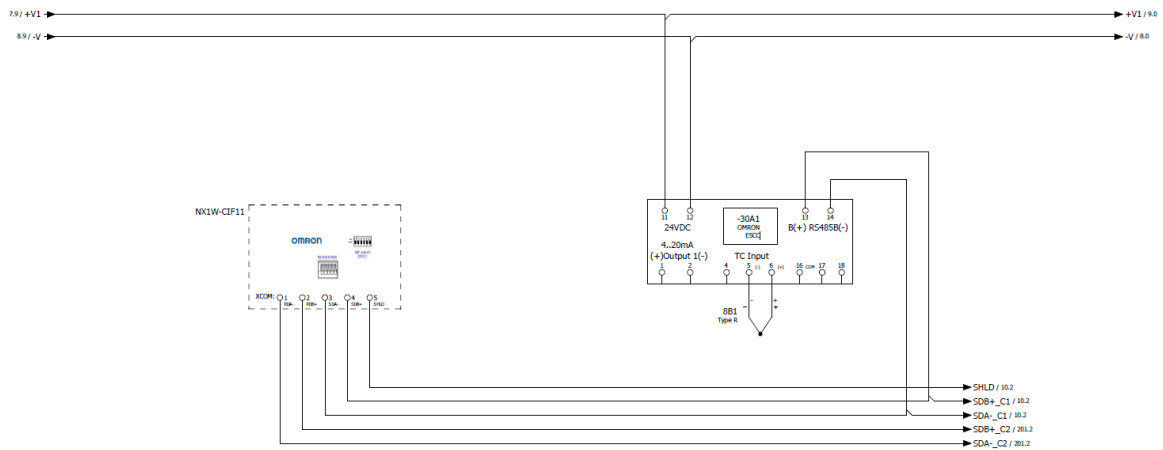


Figura 28. Comunicação entre o módulo CIF e a sonda de temperatura.

### 4.5.1. Programa de controlo

O Sysmac Studio é o *software* que fornece um ambiente integrado para configuração, programação e manutenção de controladores de automação industrial, incluindo as Unidades de CPU da série NJ / NX, o PC Industrial da série NY, o EtherCAT Slave e a HMI.

O programa de controlo é dividido em várias subsecções para permitir um programa mais bem estruturado e de fácil perceção como é mostrado na Figura 29.



Figura 29. Seções do programa de controlo rotomoldagem

#### 4.5.1.1. Geral

A seção “Geral”, tem como finalidade iniciar o processo e ainda a gestão da sinalização da máquina.

Assim que o processo é iniciado existe um pequeno temporizador geral para dar um atraso na inicialização do programa. Esta boa prática de programação tem como efeito garantir que os sistemas de comunicação estejam prontos para poder iniciar o processo.

Para além desta existe a gestão do “arranque a frio”. Este comando tem a função de colocar um atraso para proceder ao aquecimento inicial ao forno se o operador o desejar. Pois caso seja a primeira vez que este seja utilizado, garante que o forno tenha a temperatura indicada quando o processo iniciar.

Nesta seção é gerida os sinalizadores de alarme e dos carros em ciclo automático ativo. O sinalizador de alarme é ativo se existir algum alarme presente. O sinalizador de carro em automático está ativo se o ciclo estiver em execução e quando acaba o ciclo o sinalizador fica intermitente.

#### **4.5.1.2. Gestão de Programas**

Nesta divisão é efetuada a gestão da receita que está ligada a supervisão. Ou seja, verifica em cada carro qual é o ciclo ativo e calcula o tempo total do forno. Este tempo é calculado consoante a soma de cada patamar. O cálculo dos tempos totais para o fim de cada patamar é feito pela soma de cada patamar com o tempo do arranque a frio.

#### **4.5.1.3. Forno**

A gestão do aquecimento do forno é feita nesta seção.

Quando as portas da estufa fecham o ventilador de recirculação é ativado. Este ventilador tem o objetivo de permitir a circulação de ar dentro da estufa, possibilitando um aquecimento da estufa mais homogéneo.

O queimador tem duas formas de ligar: quando as portas da estufa fecham e se o operador fizer um arranque a frio. O arranque a frio permite que o forno mantenha uma temperatura estável. Quando o molde entra dentro da estufa o ar já está quente, o que possibilita uma redução no tempo de ciclo.

A temperatura dentro da estufa é medida através de uma sonda de temperatura, que mede com uma resolução de grau centígrado.

O ventilador de exaustão é ligado sempre que o queimador é ativo. Este ventilador tem como objetivo remover o fumo existente resultante da combustão.

#### **4.5.1.4. Carros**

Nesta seção é feita a gestão dos carros que é semelhante para os carros 1 e 2. A Figura 30, abaixo representada, demonstra o fluxograma de controlo.

No início do ciclo o carro pode-se encontrar dentro ou no interior da estufa. Em seguida o carro vai para a posição forno, que corresponde ao carro dentro da estufa. Este é então colocado em rotação e é iniciado o ciclo de cozedura. Durante a cozedura, consoante a escolha do utilizador, o carro pode inverter a rotação. No fim da cozedura a porta é aberta e o carro vai para a posição de arrefecimento forçado com um tempo específico. Por último a porta da estufa é fechada.

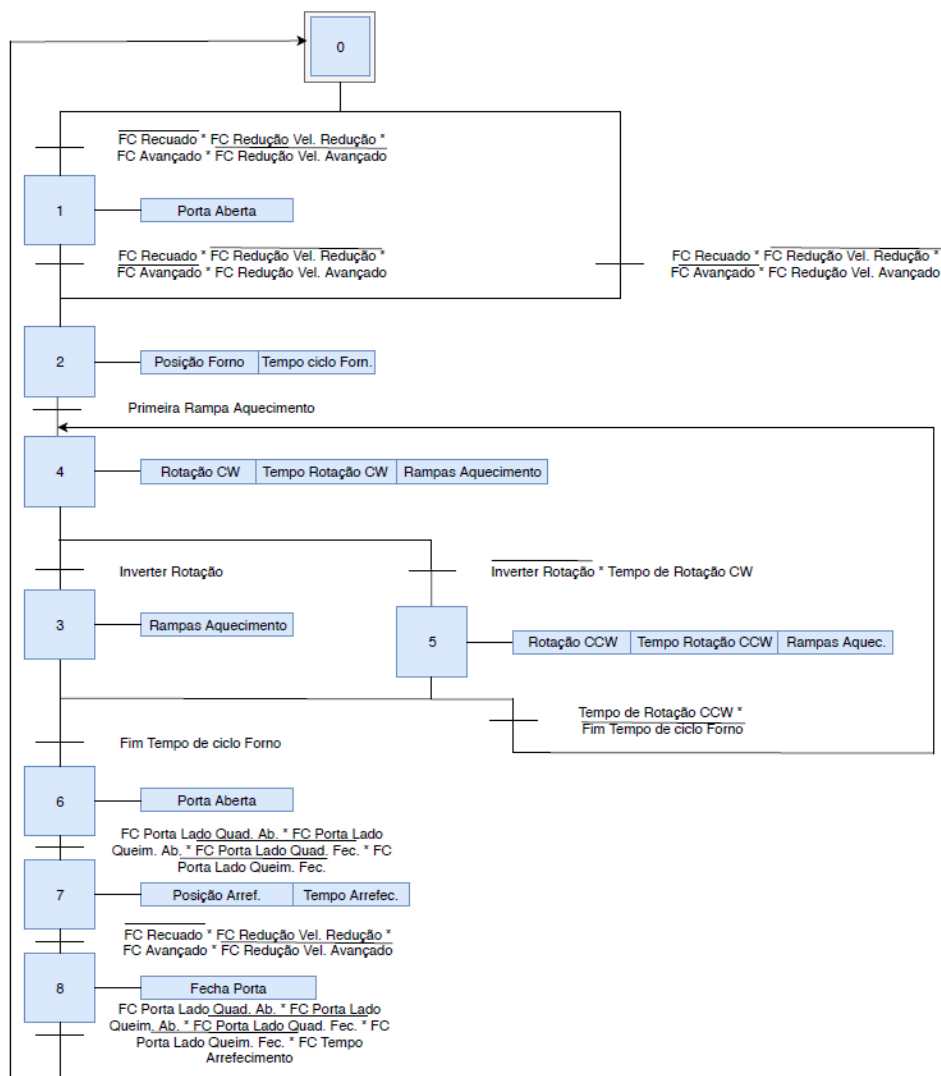


Figura 30. Fluxograma da gestão dos carros

#### 4.5.1.5. Balanceamento

Durante a rotação do molde é necessário que este esteja centrado, para que não existam oscilações. A verificação da descompensação é feita a partir da corrente consumida pelos motores durante a rotação do eixo. Se existir um aumento de corrente é possível verificar, que o molde está descompensado.

A correção da descompensação é feita manualmente pelo operador, a partir dos pesos que se encontram no eixo principal. O valor do peso é feito por tentativa e erro.

#### 4.5.1.6. RS485

Nesta seção são geridas as comunicações através do protocolo *Modbus* (RTU) entre o variador e a sonda de temperatura. Cada variador é definido como um “nó” da rede. Os “nós” são endereçados nos parâmetros do variador.

No autômato é configurado o *devicePort*, que indica qual o módulo de comunicação usado e a porta correspondente. A Figura 31 apresenta a especificação do módulo de comunicação.

```
1 DevicePort.DeviceType:=_eDEVICE_TYPE#_DeviceOptionBoard;
2 DevicePort.OptBoard:=CIF11;
3 DevicePort.PortNo:=1;
4
```

Figura 31. Especificação do módulo de comunicação.

A leitura e escrita dos “nós” é feita sequencialmente para garantir que todos são lidos sem sobrecarregar a comunicação (Figura 32).

```
1 Array_Nos_Ativos[1]:=TRUE;
2 Array_Nos_Ativos[2]:=TRUE;
3 Array_Nos_Ativos[3]:=TRUE;
4 Array_Nos_Ativos[4]:=TRUE;
5 Array_Nos_Ativos[5]:=TRUE;
6 Array_Nos_Ativos[6]:=TRUE;
7 Array_Nos_Ativos[7]:=TRUE;
8 Array_Nos_Ativos[8]:=TRUE;
9
10  Array_Nos_Ativos[10]:=TRUE; (*Controlador de Temperatura Principal*)
11  Array_Nos_Ativos[11]:=FALSE; (*Controlador de Temperatura de Alarme Queimador*)
12
```

Figura 32. Ativação de cada nó.

Os variadores possuem três tipos de dados, comando, monitorização e de transmissão.

Os dados de comando podem ler e gravar registos, os *bits* que não são usados devem ser escritos com 0. Os dados de monitorização são somente de leitura. Os registos de transmissão são enviados para todos os escravos ao mesmo tempo, o que sobrecarrega a rede.

Os registos de comandos e de monitorização são os únicos aplicados neste projeto. Para efetuar a comunicação dos tipos de dados são utilizadas funções diferentes.

A leitura dos registos nos variadores é semelhante à das sondas de temperatura (Figura 33). Para configurar o endereço de leitura consultamos as tabelas do variador para se saber quais os registos a usar (Figura 34).

```
Cmd_Leitura.Fun :=_MDB_READ_HOLDING_REGISTERS;
Cmd_Leitura.ReadAdr:=UINT#16#0020;
Cmd_Leitura.ReadSize:=7;
```

Figura 33. Exemplo de uma leitura do variador

| Register No. | Contents       |   |
|--------------|----------------|---|
| 0020H        | Drive Status   |   |
|              | bit 0          | During Run  |
|              | bit 1          | During Reverse  |
|              | bit 2          | Drive Ready   |
|              | bit 3          | Fault   |
|              | bit 4          | Data Setting Error                                      |
|              | bit 5          | Multi-Function Contact Output (terminal MA/MB-MC)       |
|              | bit 6          | Multi-Function Photocoupler Output 1 (terminal P1 - PC) |
|              | bit 7          | Multi-Function Photocoupler Output 2 (terminal P2 - PC) |
|              | bit 8 to bit D | Reserved  |
|              | bit E          | ComRef status   |
|              | bit F          | ComCtrl status  |

Figura 34. Registos utilizados para monitorização dos variadores (26)

Na sonda de temperatura para extrair um valor, definiu-se qual é a posição no *array* de comando que queremos executar e em segunda colocamos o *bit*, como é demonstrado na Figura 35.

```
Command_Data[0]:=BYTE#16#01;
Command_Data[1]:=BYTE#16#00;
```

Figura 35. Exemplo de acesso a um registo de controlo.

| Command code | Command content        | Related information                    |
|--------------|------------------------|--|
| 00           | Communications writing | 00: OFF (disabled)<br>01: ON (enabled) |
| 01           | RUN/STOP               | 00: Run<br>01: Stop                    |

Figura 36. Código de comando da sonda de temperatura (27)

#### 4.5.1.7. Alarmes

A gestão dos alarmes é executada nesta seção. Na Figura 37 é possível observar um exemplo de alarme.

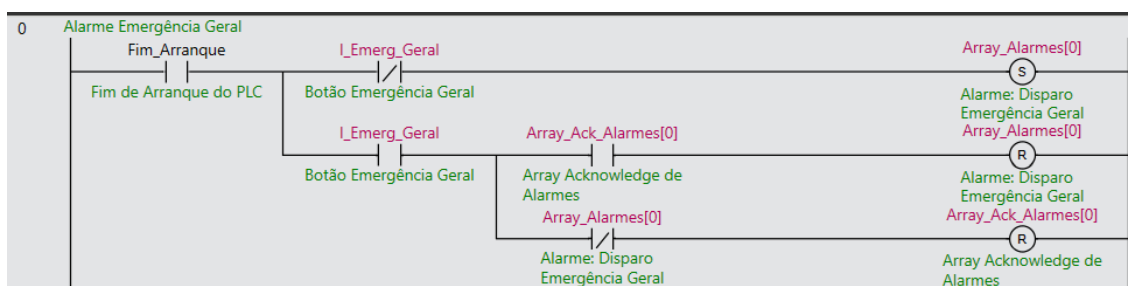


Figura 37. Exemplo de um alarme

Como mostra a Figura 37 é feita a ativação e desativação de um alarme. O alarme é ativo assim que o botão de emergência é ativo e é desativado assim que seja confirmado no *acknowledge* ou assim que o botão de emergência seja desativado.

O *acknowledge* do alarme tem a finalidade de dar o reconhecimento de alarme, isto é, o operador reconhece o alarme, mas não significa necessariamente que o alarme seja desativo. As vantagens de usar este método consistem em permitir que o processo em alarme prossiga, permitindo ainda que quando se retira o alarme temos a certeza que é feito o *reset* ao alarme.

Na Figura 38 pode-se observar a gestão dos alarmes. Um ciclo de repetição percorre o *array* dos alarmes e verifica se existe algum ativo. Caso isto aconteça é colocada uma variável auxiliar a *true*, neste caso “Alarme\_Presente”.

```

61 Ciclo de Verificação de Alarmes
1  Alarme_Presente:=FALSE;
2  Alarme_Nao_Reconhecido:=FALSE;
3  FOR i:=INT#0 TO INT#239 BY INT#1 DO
4  IF Array_Alarmes[i]=TRUE THEN
5      Alarme_Presente:=TRUE;
6  END_IF;
7
8  aux_novo_Alm:=Array_Alarmes[i] XOR Array_Ack_Alarmes[i];
9  aux_novo_Alm:=Array_Alarmes[i] AND aux_novo_Alm;
10
11 IF aux_novo_Alm=TRUE THEN
12     Alarme_Nao_Reconhecido:=TRUE;
13 END_IF;
14 END_FOR;
15

```

Figura 38. Gestão dos alarmes

O reconhecimento do alarme é feito através de operações de portas lógicas. O alarme não é reconhecido se a posição do *array* do alarme e a posição do *array* do *acknowledge* forem diferentes ou se posição do *array* do alarme for igual ao novo alarme.

Na Figura 39 é observado a gestão do *acknowledge* assim que o botão de *reset* dos alarmes é ativo. Percorre-se o *array* do *acknowledge* é colocado a posições a *false*.

```

Acknowledge dos Alarmes
1 IF I_Bot_Reset_Alarmes=TRUE THEN
2   FOR i:=INT#0 TO INT#239 BY INT#1 DO
3     Array_Ack_Alarmes[i]:=FALSE;
4   END_FOR;
5 END_IF;

```

Figura 39. Gestão *acknowledge*

## 4.6. Autómatos de segurança

O autómato de segurança é responsável por todos os sistemas de segurança do processo, como por exemplo barreiras de segurança, botoeiras de emergência e por todos os equipamentos de proteção do processo.

Este equipamento deve de atender a um conjunto rigoroso de normas internacionais, especificamente a IEC 61508. Esta norma é específica a “Segurança funcional de sistemas elétricos/eletrônicos/eletrônicos programáveis referidos à segurança”, sendo formada pelas seguintes partes:

- Parte 1: Introdução ao conceito de segurança funcional, resumo das normas da série IEC 61508;
- Parte 2: Exigências quanto aos sistemas elétricos/eletrônicos/eletrônicos programáveis relativos à segurança;
- Parte 3: Exigências quanto ao *software*;
- Parte 4: Termos e abreviações;
- Parte 5: Exemplos para verificação da classificação da integridade de segurança;
- Parte 6: Diretriz de aplicação para Parte 2 e Parte 3;
- Parte 7: Instruções de aplicação sobre métodos e medidas.

Em resposta à norma, os fornecedores deste equipamento adotaram um conceito de “Camadas de proteção”, para atingir o nível de proteção industrial o equipamento tem de possuir uma classificação de segurança “SIL3”.

O autômato de segurança implementado neste processo é da marca Omron, o modelo é g9sp-n10s (Figura 40). A comunicação usada nestes controladores é baseada em Ethernet (protocolo Omron FINS), EtherNet / IP ou serial (RS-232). Este dispositivo é certificado ISO 13849-1 e pela norma IEC61508 (SIL3).



Figura 40. Autômato: G9SP-N10S

Características do autómato de seguranças:

Tabela 3. Especificações do autómato: G9SP-N10 S

|                        |           |
|------------------------|-----------|
| Entradas de seguranças | 10        |
| Saídas de segurança    | 4         |
| Saídas de teste        | 4         |
| Saídas Normais         | 4         |
| Peso                   | 290g max. |

No sistema de segurança deste processo é aplicado o *safety light curtains* ou cortinas de luz ou barreiras de segurança. São dispositivos em que um transmissor fotoelétrico projeta um conjunto de feixes de luz infravermelha sincronizada e paralela a uma unidade recetora. A deteção é feita quando um objeto interrompe um ou mais feixes no campo de deteção (28).

A Figura 41 mostra o esquema de ligação das entradas do autômato de Segurança.

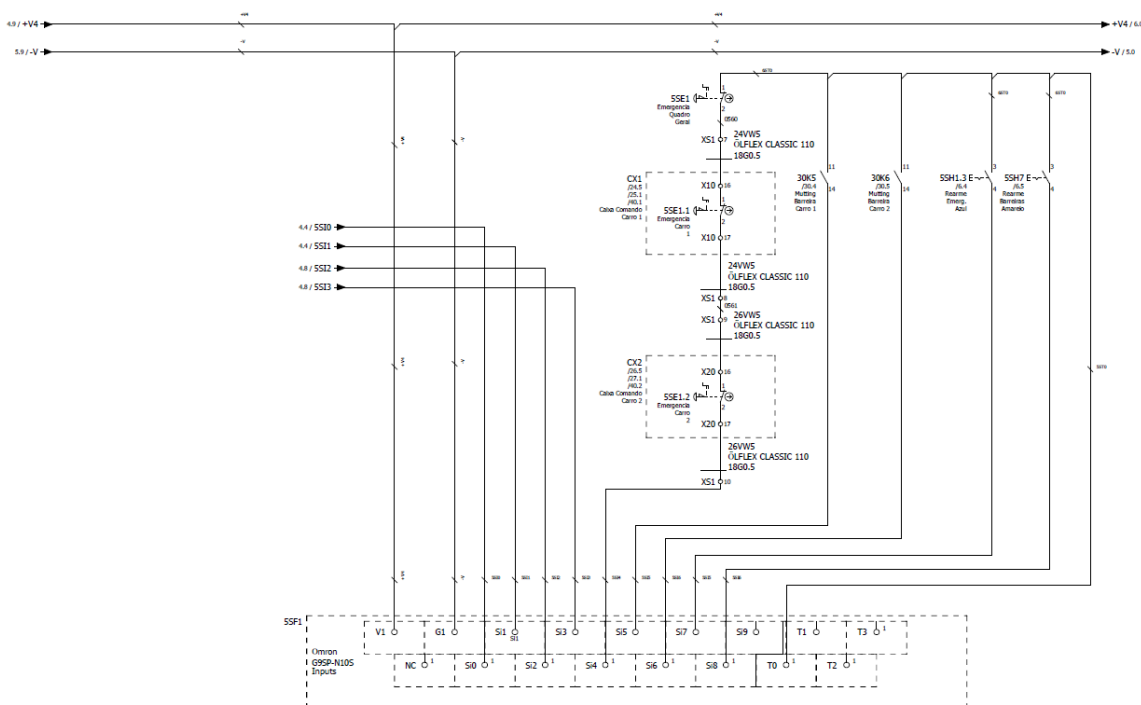


Figura 41. Esquema de entradas no autômato de segurança.

Neste esquema de ligações verifica-se que o autômato de segurança é alimentado de forma dedicada pela designa-se por V4.

Tabela 4. Entradas do autômato: G9SP-N10 S

|     |                                |
|-----|--------------------------------|
| Si0 | Barreiras de segurança Carro 1 |
| Si1 | Barreiras de segurança Carro 1 |
| Si2 | Barreiras de segurança Carro 2 |
| Si3 | Barreiras de segurança Carro 2 |
| Si4 | Emergência dos carros e geral  |
| Si5 | <i>Muting</i> Carro 1          |
| Si6 | <i>Muting</i> Carro 2          |
| Si7 | Rearme das Seguranças          |
| Si8 | Rearme das Barreiras           |

A T0 é uma saída de teste que emite um pulso. Este controla e monitoriza as entradas de segurança para verificar o pulso, é se possui um retorno sem alterações temporais. Caso contrário, pode existir um curto-circuito.

Tabela 5. Saídas do autómato: G9SP-N10 S

|     |                           |
|-----|---------------------------|
| SO1 | Entrada do autómato       |
| SO2 | Entrada do autómato       |
| SO3 | Entrada do autómato       |
| 00  | Sinalizador de Barreira   |
| 01  | Sinalizador de Emergência |

Na Figura 42, é possível observar o esquema de ligação das cortinas de luz neste processo.

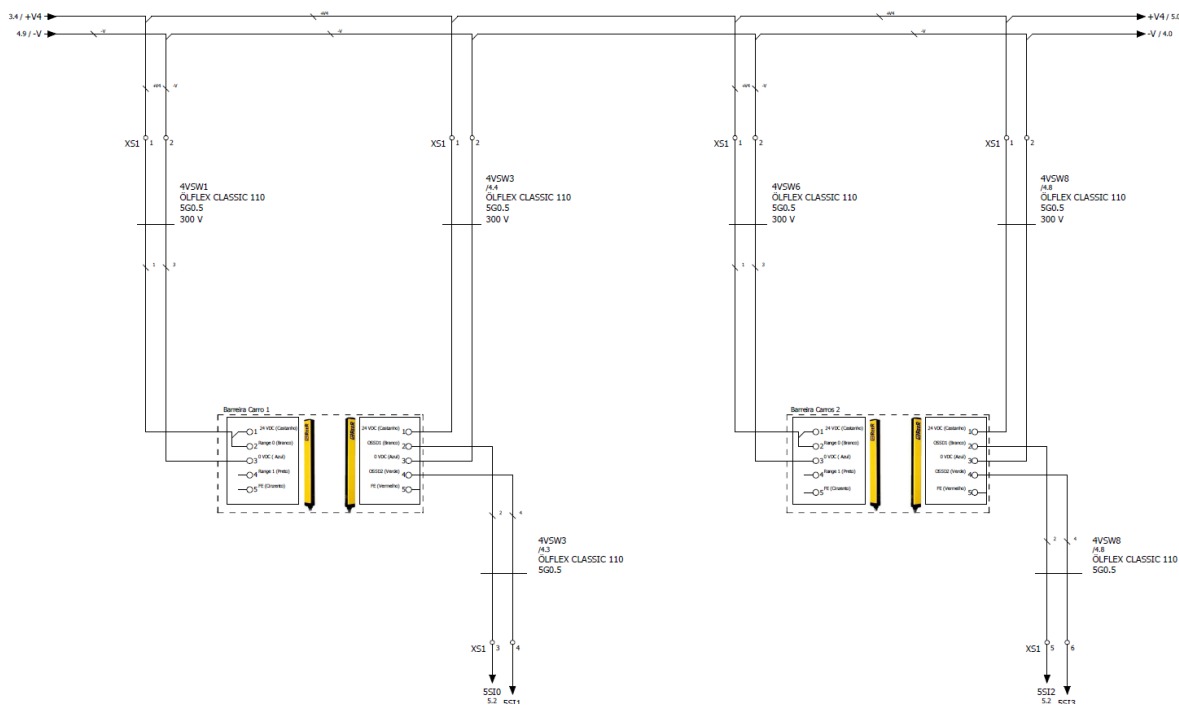


Figura 42. Esquema de Ligação das cortinas de luz.

#### 4.6.1. Programação do autômato de segurança

A programação deste controlador foi feita pelo programa, *G9sp Configurator*. A linguagem de programação é desenvolvida através de portas lógicas e por blocos de funções específicas deste equipamento de segurança (29).

A lógica de programação implementada neste processo consiste em: assim que a máquina inicia-se o operador tem de fazer o rearme das barreiras e da emergência. De seguida, sempre que uma barreira seja ativa irá aparecer ativo um sinalizador de zona e outro de barreiras. No caso em que o operador tenha de ativar as barreiras durante o processo, este pode acionar o botão de *muting* que desativa temporariamente os sinais de entrada das barreiras.

Na Figura 43 mostra-se um esquema com a programação feita para este processo.

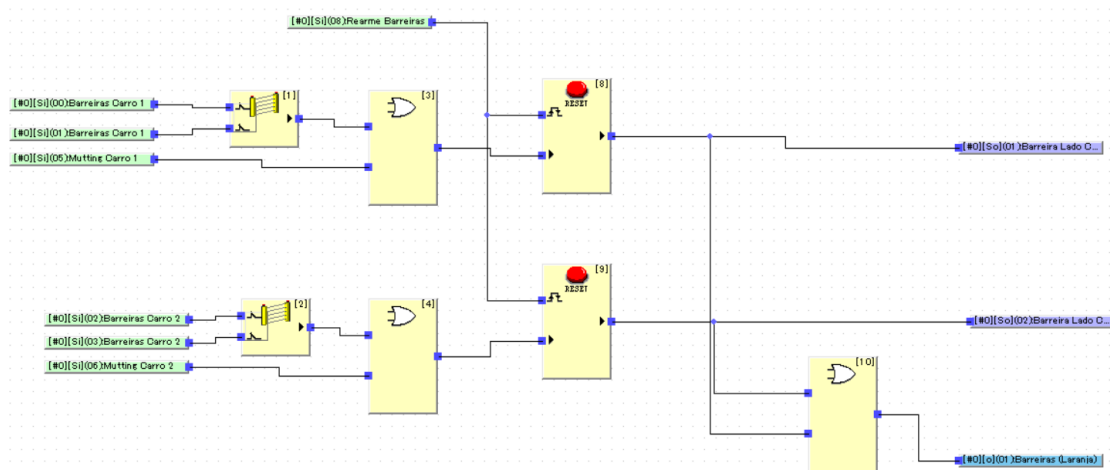


Figura 43. Programação das Barreiras de segurança.

O esquema de programação para a emergência geral é semelhante ao das barreiras de segurança. A única diferença está em não ser usado o *muting*.

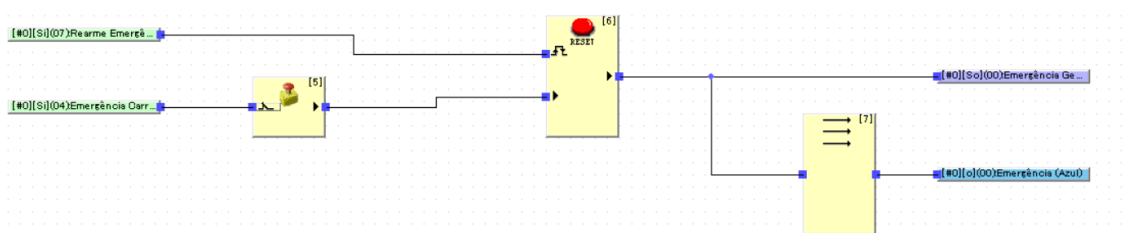


Figura 44. Programação do botão de emergência

## 4.7. Sistema de supervisão

Neste tópico é abordado o desenvolvimento do sistema de supervisão assim como as suas funções. As secções abordadas são:

- Página principal: Forno;
- Registo de Produção;
- Relatório de produção;
- Gestão das Receitas;
- Alarmes;
- Gestão do Balanceamento.

### 4.7.1. Página principal: Forno

A Figura 45, apresenta o *front end* de monitorização do funcionamento da máquina.

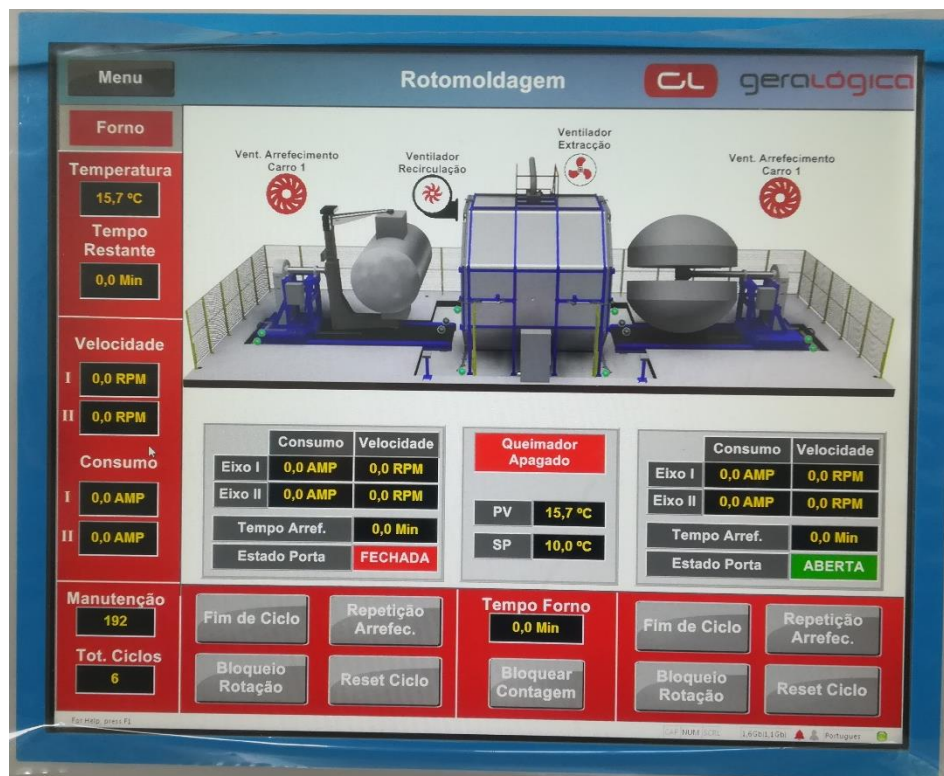


Figura 45. Ecrã da página do Forno

Esta página tem a função de monitorização do processo em tempo real. Este é constituído por uma panóplia de campos para verificação do estado de execução de cada constituinte da máquina desde os carros, os ventiladores de recirculação, o ventilador de extração, do queimador e o estado das portas.

Na indicação dos carros é possível observar qual o consumo instantâneo e a velocidade de cada eixo. Permite ainda verificar qual o carro ativado e a localização dos carros através dos fins de curso.

No queimador existe a indicação da temperatura em tempo real PV (*present value*) o valor a atingir SP (*set point*) sendo possível verificar o estado do mesmo.

O operador pode interagir com o processo através dos diferentes botões, que são:

- Fim de Ciclo - Tem a finalidade acabar com o ciclo em andamento;
- Repetição Arrefecimento - Tem como função fazer repetir o ciclo de arrefecimento sempre que este termina;
- Bloqueio da Rotação - Vai bloquear a rotação dos eixos I e II;
- *Reset* Ciclo - Reinicia o ciclo em andamento;
- Bloquear Contagem - Ativa o bloqueio de tempo do molde no forno.

#### **4.7.2. Registo de produção**

O registo de produção é um complemento ao processo, este possibilita a realização do histórico da máquina.

Este registo de produção é feito por quatro janelas separadamente: moldes; materiais; produtos e classificação. Através do objeto do Movicon, “*TAB Group*”, foi possível a junção dessas páginas.

As páginas apresentam as matérias, moldes, produtos e a classificação, como mostra a Figura 46.



Figura 46. Página de registo de produção

A lógica de programação dos botões é semelhante em quase todas as páginas à exceção da página da classificação do molde.

A função de cada botão é a seguinte:

- Botão de Guardar - Permite guardar um novo registo na base de dados ou alterar um registo existente;
- Botão de Eliminar - Elimina o registo que estiver selecionado;
- Botão de Cancelar - Limpa todos os campos da janela para uma nova.

#### 4.7.2.1. Materiais

Esta janela (Figura 46) permite a criação dos materiais constituintes dos moldes, logo está interligada com a janela dos “moldes”. Nesta janela podemos inserir, alterar, eliminar matérias da base de dados.

A janela “material” tem uma particularidade em relação as outras páginas. É possível observar os materiais já inseridos na tabela através do objeto “Grid” do Movicon.

#### 4.7.2.2. Moldes

Nesta página são criados os moldes aplicados nas receitas. Assim, quando o operador realizar um relatório de produção sabe qual o molde utilizado em cada ciclo do processo.

Na janela “moldes” pode ser inserido o fornecedor, a descrição, os materiais, o peso desses materiais, o tempo e a temperatura de cozedora (Figura 47).

Figura 47. Página Moldes.

#### 4.7.2.3. Classificação

Nesta janela é possível classificar os moldes consoante o tipo de defeito. Para isso é feita uma filtragem do dia em que foi feito o ciclo, sendo de seguida inserido o carro e o número do ciclo.

Podem surgir dois tipos de problemas: *rejeitado* e *perda de tempo*. O modo *rejeitado* é quando o produto está destruído, a *perda de tempo* é usado sempre que o processo demorar mais tempo que o previsto.

A *classe de defeito* é uma listagem que varia consoante o tipo de problema. Em termos de programação, esta listagem é acedida a partir de um ficheiro de texto. Em seguida seleciona-se o molde que está danificado.

A nível de programação é feito um filtro da data. A partir desta filtragem são atualizados os valores da *CombList* do carro e do número de ciclo.

Na Figura 48, pode-se observar o *templates* da página da classificação dos defeitos de produção.

Figura 48. Página Classificação.

### 4.7.3. Relatório de produção

O relatório de produção permite à organização entender as falhas, perdas de tempo e moldes usados durante as receitas para uma melhor gestão e menor perda de tempo. Por isso, esta página tem uma janela onde é possível a observação dos registos selecionados. O operador tem diferentes formas de fazer uma filtragem dos registos através da data, intervalos de duas datas, uma data e um carro ou apenas a partir do carro selecionado, como é demonstrado na Figura 49.



Figura 49. Página relatório de produção

No caso de o operador selecionar a data, esta é dividida em variáveis separadas em ano, mês e dia. A data passa inicialmente por uma verificação se o ano é anterior a 2018, se o mês é menor que 1 ou superior a 12 e o dia se é inferior a 1 e superior a 31. Depois desta verificação, existe a concatenação de um 0 ao dia caso este seja inferior a 9, como demonstra a Figura 50.

```
data_1 = CStr( "0"&Filtro_Dia & "-" & "0"&Filtro_Mes & "-" & Filtro_Ano)
```

Figura 50. Concatenação da data.

Em seguida é verificado se a data inserida pelo operador existe na base de dados. Para isso aplicamos a instrução *select count* (Figura 51) que conta todas as ocorrências da data selecionada.

```
SQLText = "Select COUNT (D_I_P_C) FROM Gestao_Classe WHERE D_I_P_C = '" & data_1 & "'"
```

Figura 51. Instrução *Select Count*.

A filtragem é feita pela instrução *select*. As condições para a seleção dos registos dependem da *checkbox* que o operador selecionar. Assim que é feita a filtragem é guardada numa variável auxiliar o filtro aplicado, como é mostrado na Figura 52. Esta informação é importante para a realização do relatório de produção.

```
SQLText = "Select Ca_C,N_C,Re_C,D_I_P_C,H_I_P_C,H_F_P_C,H_I_S_C,H_F_S_C, H_I_T_C,H_F_T_C, " & _
"M_1_C,M_2_C,M_3_C,M_4_C,M_5_C,M_6_C,M_7_C,M_8_C,M_9_C,M_10_C, D_C,T_D_C " & _
"FROM Gestao_Classe WHERE D_I_P_C = '" & data_1 & "'"
rs.Open SQLTEXT, dbCon
Sel_T_Filtro="Filtro Data: "&data_1&""
```

Figura 52. Exemplo de uma filtragem.

Os registos seleccionados pela filtragem são apresentados numa tabela através do objeto “*grid*” (Figura 53). Este objeto permite que os conteúdos dos registos da base de dados sejam exibidos no formato de tabela. Os dados podem ser exibidos de forma parcial ou totalmente extraídos, de acordo com a filtragem definida.

|   | Molde 7 | Molde 8 | Molde 9 | Molde 10 | Defeito     | Tipo de Defeito |
|---|---------|---------|---------|----------|-------------|-----------------|
| 1 | Teste_7 | Teste_8 |         |          | Ok          |                 |
| 2 | Teste_7 | Teste_8 |         |          | Ok          |                 |
| 3 | Teste_7 | Teste_8 |         |          | Rejeitado   | Marcas do gel   |
| 4 | Teste_7 | Teste_8 |         |          | Perda_Tempo | vczvx           |
| 5 | Teste_7 | Teste_8 |         |          | Ok          |                 |
| 6 | Teste_7 | Teste_8 |         |          | Ok          |                 |

Figura 53. Tabela do Relatório de Produção

A construção das colunas da tabela é usada a biblioteca correspondente a “*grid*” através do comando *AddColumn*.

A representação das linhas é um pouco mais complexa. Numa primeira fase extrai-se os registos da base de dados com a função de SQL “*Select*” e em seguida através do objeto *ADO Field*, que contém as informações sobre o campo específico segundo a seleção feita.

O *ADO Field* é uma coleção do objeto *ADO Recordset*. Este é utilizado para armazenar um conjunto de registos de uma tabela da base de dados. Um objeto *Recordset* consiste em registos das colunas (campos).

No seguimento da filtragem, todos os registos que são seleccionados são copiados para uma base de dados auxiliar para a realização de um relatório. Este método foi aplicado para haver uma simplificação no processo de realização do relatório.

Para a criação do relatório de produção depois da filtragem dos registos, os valores são guardados numa tabela auxiliar, uma vez que no programa usado para fazer o *template* do relatório, as variáveis tinham de estar inseridas numa tabela.



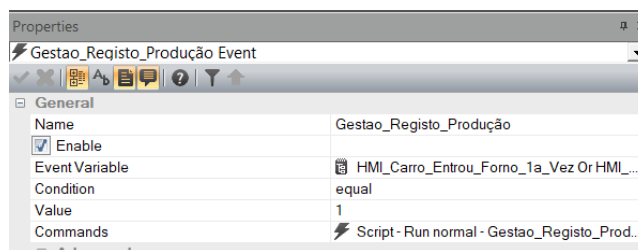


Figura 56. Definir um evento

O funcionamento deste *script* é o seguinte. Quando o carro entra dentro do forno numa primeira vez é guardada a hora, o dia, o número do carro e incrementado o número de ciclo. Consoante o carro que esteja a ser usado, carro 1 ou carro 2, é criado um registo na base de dados. Quando é alterado o carro em utilização, o valor do número de ciclo é limpo.

Assim que o carro que sai o registo da base de dados é atualizado com o valor da hora final do ciclo.

Se o mesmo carro reentrar no forno o registo da base de dados é atualizado com o valor da data e hora inicial. Quando o carro sai é atualizado o registo com a data e hora final do ciclo.

#### 4.7.4. Gestão das Receitas

A página das “receitas” é a mais importante ao nível de interface humano-máquina. Nesta página é realizada a gestão dos tempos, duração dos ciclos, velocidades dos moldes e algumas configurações de temperatura da estufa.

A gestão das receitas é dividida entre os dois carros, no entanto estas podem ser utilizadas por ambos.

A Figura 57 mostra o *template* da gestão das receitas. Este é feito por cinco patamares, onde se pode alterar a temperatura do forno, a duração do tempo de ciclo e a velocidade do eixo 1 e 2. Esta máquina está ainda dotada de uma funcionalidade de reaquecimento do molde necessário em certos polímeros. Para isso, o operador pode colocar valores no ciclo duplo ou triplo.

Figura 57. Gestão das Receitas.

Algumas das configurações mais importantes na gestão da receita são:

- Temperatura em vazio - esta serve para manter a temperatura quando o forno está em vazio sem ter molde inserido;
- Temperatura máxima - É a temperatura máxima do forno para ativar o alarme de excesso de temperatura;
- Alarme diferente de temperatura - Este alarme é ativo quando a temperatura de *set point* excede o tempo configurado;
- Atraso de início de rotação I e II - Este atraso é o tempo que o molde tem para começar a rodar assim que entra dentro do forno.

As receitas não são geridas pela base de dados para haver uma maior facilidade de transpor os valores das variáveis para o autómato. Para isso utilizou-se ficheiros de texto aplicando comandos específicos de *vb.net* para efetuar a leitura, escrita e apagar dados desses ficheiros.

As variáveis usadas nesta gestão estão divididas em dois tipos, relacionadas com a edição dos campos que estão dentro de uma estrutura para ser mais fácil a sua consulta. Estas são variáveis auxiliares do Movicon. As outras variáveis são diretamente extraídas do autómato. Esta divisão é feita para que não sejam enviadas variáveis para o autómato erradamente. Um exemplo disso é o caso de um operador colocar um valor num campo e

esse valor ser logo enviado para o autómato sem que seja dada a confirmação para a alteração desse valor.

#### 4.7.4.1. Gravação da Receita

A gravação da receita é iniciada assim que o operador coloca o nome da receita. Esse nome é verificado na base dados, para não existir duplicação de nomes. Se existirem é apresentado uma *pop-up* com a mensagem “Existe uma receita com esse nome”. Na Figura 58 é demonstrado o fluxograma da gravação das receitas

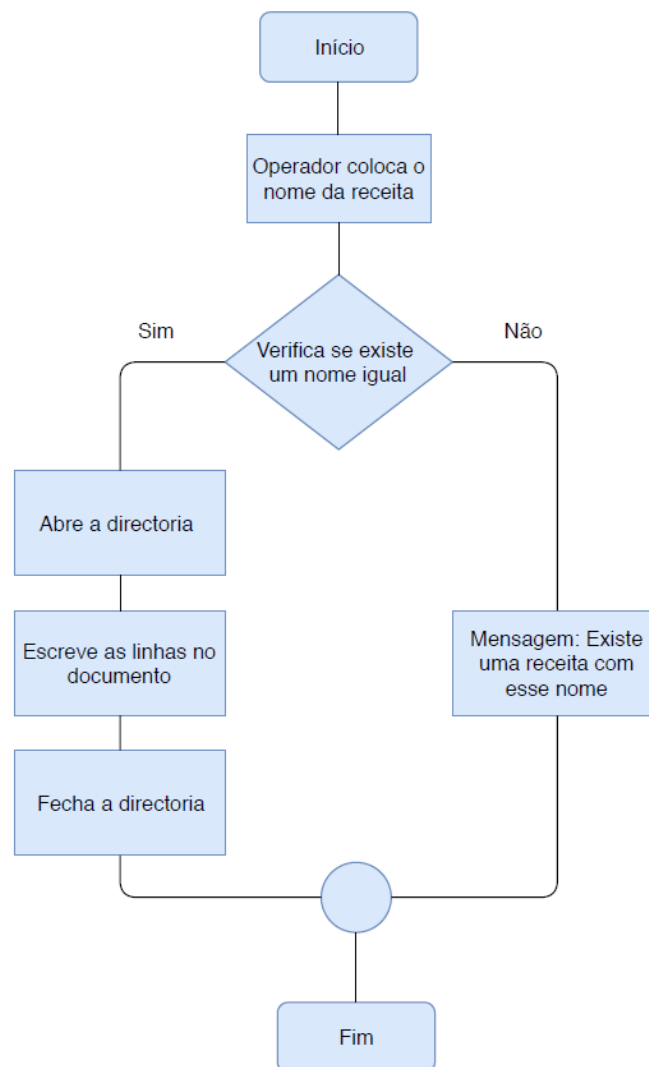


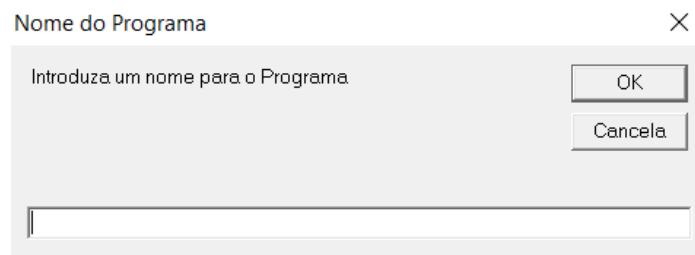
Figura 58. Fluxograma de gravação da receita.

Se o nome não existir é aberta a diretoria do ficheiro texto e escreve as linhas com o valor das variáveis a ser guardadas, através da função “*write*” (Figura 59). Por fim, é fechada a diretoria.

```
Open "C:\Users\Micael Rodrigues\Desktop\Relatorio_Estagio\Rotomoldagem\Rotomoldagem_Supervisao_1.9\Receitas\" & [Prog_Edicao:Nome] & ".csv" For Output As #  
Write #1, [Prog_Edicao:Nome], "Nome do Programa"  
Write #1, [Prog_Edicao:Cod_Molde_1], "Código Molde 1"  
Write #1, [Prog_Edicao:Cod_Molde_2], "Código Molde 2"  
Write #1, [Prog_Edicao:Cod_Molde_3], "Código Molde 3"  
Write #1, [Prog_Edicao:Cod_Molde_4], "Código Molde 4"  
Write #1, [Prog_Edicao:Cod_Molde_5], "Código Molde 5"  
Write #1, [Prog_Edicao:Cod_Molde_6], "Código Molde 6"  
Write #1, [Prog_Edicao:Cod_Molde_7], "Código Molde 7"  
Write #1, [Prog_Edicao:Cod_Molde_8], "Código Molde 8"  
Write #1, [Prog_Edicao:Cod_Molde_9], "Código Molde 9"  
Write #1, [Prog_Edicao:Cod_Molde_10], "Código Molde 10"
```

Figura 59. Escrita no ficheiro texto

A Figura 60 mostra um exemplo da *pop-up* para a colocação do nome da receita que aparece assim que o operador aciona o botão de guardar uma receita.

Figura 60. *Pop-up*: Colocação do nome da receita.

#### 4.7.4.2. Eliminar uma receita

A lógica usada para a eliminação das receitas é representada através do fluxograma apresentado na Figura 61.

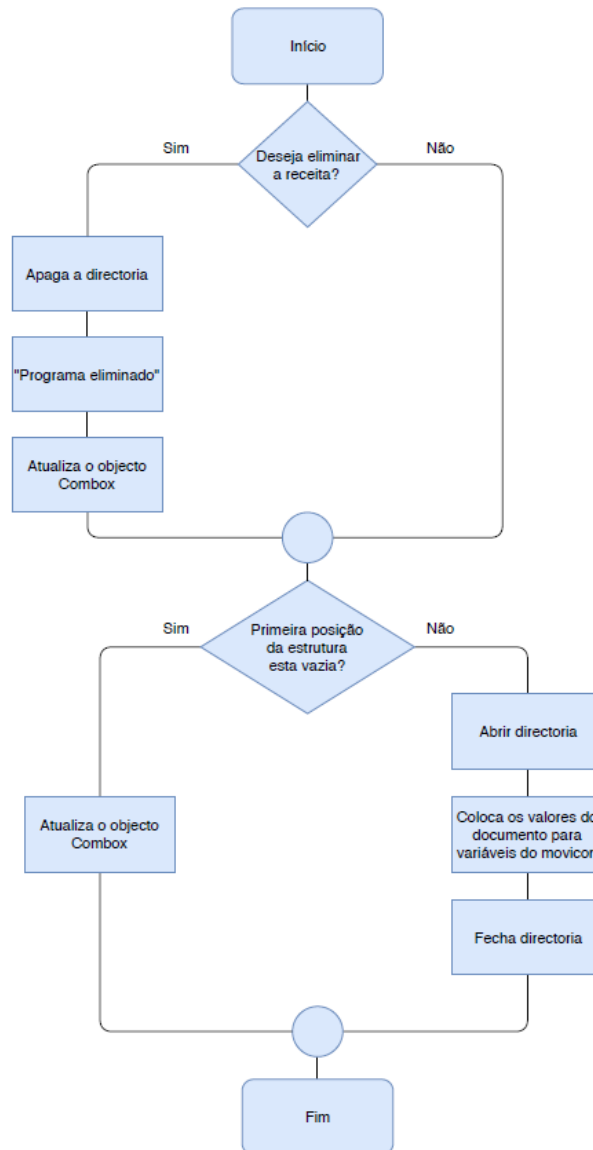


Figura 61. Fluxograma de eliminar receita

Assim que a receita é eliminada a directoria é apagada e atualiza a “Combox”. É feita uma verificação da posição da estrutura e ativa a directoria que esteja a seguir na pilha. Assim todos os valores das variáveis da receita ativa são transpostos para as variáveis auxiliares do Movicon para serem colocadas nos campos das receitas.

#### **4.7.4.3. Cancelar edição de receita**

Este botão pretende desfazer as alterações que o operador possa fazer em qualquer campo de uma receita anteriormente guardada. Para isso, sempre que o botão é pressionado é acedida a diretoria da receita ativa e são copiadas as variáveis do ficheiro para as do Movicon. Deste modo atualizam-se os campos para os valores da receita anterior das alterações.

#### **4.7.4.4. Upload e Download**

O envio de dados para o autómato é feito pelo botão com a imagem de uma seta virada para o autómato, que vai transferir os valores da receita atual para o autómato.

O envio de dados para o *PainelPC* é efetuado pelo botão com a seta virada para o Monitor, esta vai transferir os valores que estão ativos no autómato e representá-los no *PainelPC*.

#### 4.7.4.5. Configurações avançadas

O próximo botão é o das configurações avançadas. Este abre uma janela com estas configurações, Figura 62. Estas não são guardadas numa receita. São configurações próprias da máquina com as velocidades manuais e automáticas da máquina para os diferentes carros e configurações de temperatura e tempos do forno.



Figura 62. Janela das configurações avançadas.

#### 4.7.5. Alarmes

A gestão dos alarmes é crucial para um processo como este, devido à possibilidade de existirem velocidades de rotação dos eixos excessivas, as temperaturas do forno muito altas, disparos dos disjuntores, falhas na comunicação, falhas nos fins de curso, falhas nas barreiras de seguranças entre muitos outros aspetos onde é importante a visualização dos alarmes.

No autómato, os alarmes estão guardados dentro de um *array* do tipo booleano com 240 posições. Cada posição corresponde a um único alarme (Figura 63).

Comment Settings

| Name            | Comment                                    |
|-----------------|--|
| ▼ Array_Alarmes | Alarme:                                    |
| [0]             | Disparo Emergência Geral                   |
| [1]             | Falta de Fase                              |
| [2]             | Disparo Disjuntor 24VDC Entradas           |
| [3]             | Disparo Disjuntor 24VDC Saídas             |
| [4]             | Disparo Disjuntor 24VDC Seguranças         |
| [5]             | Disparo Disjuntor 24VDC Freios             |
| [6]             | Disparo Disjuntor 230VAC Comandos          |
| [7]             |  |
| [8]             |  |
| [9]             |  |
| [10]            | Variador Ventilador Recirculação Forno     |
| [11]            | Variador Ventilador Extração Forno         |
| [12]            | Deteção de Chama Queimador do Forno        |
| [13]            | Queimador Bloqueado                        |
| [14]            | Sonda de Temperatura Danificada            |
| [15]            | Temperatura Máxima Forno                   |
| [16]            | Diferencial Temperatura Forno              |
| [17]            | Queimador Ligado com Porta Aberta          |
| [18]            |  |
| [19]            |  |
| [20]            | Falha Comunicação Controlador Temperat     |
| [21]            | Falha Comunicação Variador Ventilador Reci |
| [22]            | Falha Comunicação Variador Ventilador Extr |
| [23]            |  |

Figura 63. Array dos Alarmes do autómato.

O Movicon tem uma ferramenta própria para a gestão dos alarmes. Esta trata os alarmes como objetos do projeto o que permite indicar o nome e a relação com as variáveis de cada alarme.

Na Figura 64 é possível observar um exemplo da inserção de um alarme no Movicon. Os alarmes estão separados por zonas onde a divisão é feita por *threshold* correspondendo à condição da ativação do alarme.

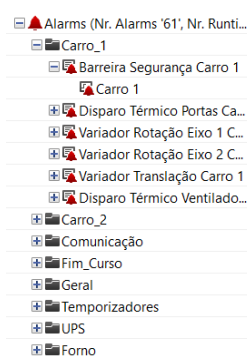


Figura 64. Exemplo de Alarmes

Na Figura 65, mostra como é realizada a ativação de um alarme que é executada se existir uma condição associada definida pelo *threshold*.

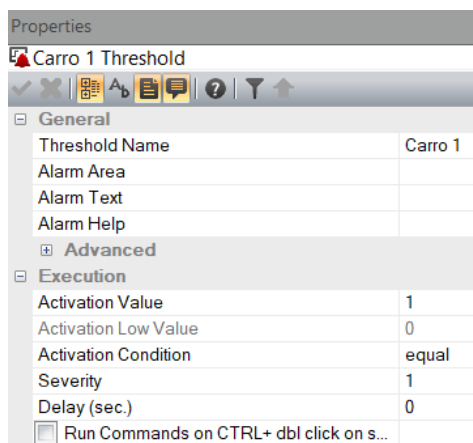


Figura 65. Exemplo da aplicação do "Threshold"

Como mostra a Figura 65 na seção "Execution" é colocada a condição de ativação do alarme. Neste caso a ativação é feita quando a variável for igual a 1, o alarme ativado é a "Barreira de segurança carro 1".

Para a verificação dos alarmes é aplicado o objeto do Movicon designado por "Alarm Window", que permite visualizar todos os alarmes ativos (Figura 66).

| Descrição                       | Estado | Duração |
|---------------------------------|--------|---------|
| ▲ Variador Ventilador Recir...  | ON     |         |
| ▲ Variador Ventilador Extraç... | ON     |         |
| ▲ Sonda de Temperatura Da...    | ON     |         |
| ▲ Falta de Fase - Seguranças    | ON     |         |
| ▲ Falha Comunicação Varia...    | ON     |         |
| ▲ Falha Comunicação Varia...    | ON     |         |
| ▲ Falha Comunicação Cont...     | ON     |         |
| ▲ Disparo Térmico Ventilad...   | ON     |         |
| ▲ Disparo Térmico Ventilad...   | ON     |         |
| ▲ Disparo Térmico Portas C...   | ON     |         |
| ▲ Disparo Emergência Gera...    | ON     |         |
| ▲ Disparo Disjuntor 24VDC ...   | ON     |         |
| ▲ Disparo Disjuntor 24VDC ...   | ON     |         |
| ▲ Disparo Disjuntor 24VDC ...   | ON     |         |
| ▲ Disparo Disjuntor 24VDC ...   | ON     |         |
| ▲ Disparo Disjuntor 230VA...    | ON     |         |

Figura 66. Exemplo de uma listagem de alarme.

No funcionamento normal, o estado dos alarmes passa do estado "ON" para o "ACK". Este significa que o alarme foi visto pelo operador, mas a máquina continua em produção, mantendo um indicador de alarme. Como este processo é muito vulnerável optou-se por manter o sinal de alarme até a avaria estar mesmo resolvida. Para isso foi elaborado um script que verifica quando o alarme não está ativo, usando as propriedades do objeto, a informação é atualizada.

O Movicon permite ainda fazer um histórico ordenado cronologicamente com todos os alarmes que foram ativos. Para isso aplicamos o objeto "Log Window", Figura 67.

| EVENTO   | HORA            | UTILIZADOR | DESCRIÇÃO   |
|----------|-----------------|------------|---|
| ALARM ON | Out of interval |            | Falha Comunicação Variador Rotação Eixo 1 Carro 1 - Com_Carro 2 : Array_Alarmes_57 - Comunicação      |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Rotação Eixo 2 Carro 1 - Com_Carro 2 : Array_Alarmes_58 - Comunicação      |
| ALARM ON | Out of interval |            | Array_Alarmes_16 - Forno : Array_Alarmes_16 - Forno   |
| ALARM ON | Out of interval |            | Variador Ventilador Extração Forno - Forno : Array_Alarmes_11 - Forno                                 |
| ALARM ON | Out of interval |            | Falha Comunicação Controlador Temperatura - Com_Forno : Array_Alarmes_20 - Comunicação                |
| ALARM ON | Out of interval |            | Variador Rotação Eixo 1 Carro 1 - Carro 1 : Array_Alarmes_54 - Carro_1                                |
| ALARM ON | Out of interval |            | Variador Rotação Eixo 2 Carro 1 - Carro 1 : Array_Alarmes_55 - Carro_1                                |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Translação Carro 2 - Com_Carro 2 : Array_Alarmes_106 - Comunicação         |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Rotação Eixo 1 Carro 2 - Com_Carro 2 : Array_Alarmes_107 - Comunicação     |
| ALARM ON | Out of interval |            | Variador Translação Carro 2 - Carro 2 : Array_Alarmes_103 - Carro_2                                   |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Translação Carro 1 - Com_Carro 1 : Array_Alarmes_56 - Comunicação          |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Rotação Eixo 2 Carro 2 - Com_Carro 2 : Array_Alarmes_108 - Comunicação     |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Ventilador Recirculação Forno - Com_Forno : Array_Alarmes_21 - Comunica... |
| ALARM ON | Out of interval |            | Falha Comunicação Variador Ventilador Extração Forno - Com_Forno : Array_Alarmes_22 - Comunicação     |
| ALARM ON | Out of interval |            | Variador Rotação Eixo 1 Carro 2 - Carro 2 : Array_Alarmes_104 - Carro_2                               |
| ALARM ON | Out of interval |            | Diferencial Temperatura Forno - Forno : Array_Alarmes_16 - Forno                                      |

Figura 67. Exemplo do *Log Window*.

Este objeto tem ainda muitas outras aplicações de visualização, como por exemplo:

- Eventos do sistema (manipulação do operador no sistema);
- Diagnóstico do sistema (inicialização de projetos, *logon / logoff* dos utilizadores, mensagens em conexões de rede);
- Diagnóstico do driver de comunicação (status da comunicação).

#### 4.7.6. Gestão do Balanceamento

O balanceamento tem um papel muito importante neste processo, pois este permite que o carro esteja equilibrado com o molde, para que não existam oscilações durante a rotação do mesmo. Na Figura 68 é possível observar a gestão de balanceamento. O operador tem de colocar o molde como está indicado na imagem, em seguida selecionar o eixo que quer calibrar e executar o *start*. Assim que se dá o *start* o molde faz uma volta completa e ao mesmo tempo é observável o ângulo de rotação.

No fim da rotação o operador tem de colocar um peso no eixo que esteja selecionado para compensar, assim como a distância do eixo.



Figura 68. Gestão de Balanceamento.

## **5. Implementação do divisor angular**

Este capítulo apresenta o desenvolvimento do projeto e implementação do divisor angular, acoplado a uma máquina CNC. Este módulo permite executar a rotação de peças, consoante um ângulo pretendido pelo operador durante a maquinação de uma peça. Neste projeto, foi implementado o esquema elétrico do módulo, o programa de controlo assim com respetiva interface HMI.

## 5.1. Controlo numérico computadorizado (CNC)

O controlo numérico computadorizado (CNC) é um processo automático onde a máquina executa um programa para realização de uma tarefa sem intervenção direta do operador, por exemplo, para a fabricação de um molde ou maquinação de metais.

A Figura 69 mostra uma CNC tradicional composta por dois eixos que se encontram no centro da figura. O divisor angular desenvolvido foi aplicado uma máquina deste género.



Figura 69. CNC de um eixo

O deslocamento da ferramenta é definido por uma trajetória plana ou de espaço, estabelecidas pelas coordenadas de X, Y e Z dos pontos a percorrer. Para isso, é necessário definir a posição da ferramenta no espaço relativamente à peça e ao ponto de origem.

A Figura 70 mostra uma CNC com três eixos. Estas máquinas têm uma capacidade de maquinação muito superior as máquinas com dois eixos, pois permite a rotação automática da peça durante o processo de maquinação.



Figura 70. CNC de 3 eixos

## 5.2. Divisor angular

O divisor angular é um acessório para uma CNC, que permite aumentar a capacidade de resposta adicionando um terceiro eixo a uma máquina com dois eixos com um custo reduzido.

Este eixo é rotativo permitindo a rotação de peças durante o processo de maquinação de forma automática.

A Figura 71 mostra o divisor angular aplicado na CNC. O desenho da máquina foi desenvolvido na empresa. Observa-se a bucha onde agarra a peça a maquinar. Tem também uma regulação por ângulos de viragem do divisor.



Figura 71. Divisor Angular

A Figura 72 apresenta os modos de funcionamento da máquina, *Manual* e *Automático*. No modo automático existem os modos *Furos* e *Tabela*.

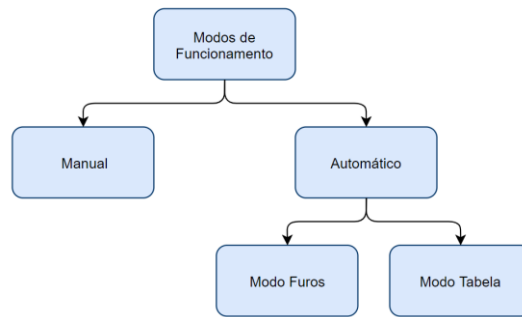


Figura 72. Modos de Funcionamento

No *Modo Furos*, o operador insere um número de furos que se divide por 360, para adquirir o ângulo de rotação.

O *Modo Tabela* é um pequeno ciclo de quatro patamares onde é inserido o ângulo da rotação e o número de repetições.

Na Figura 73 é perceptível o quadro elétrico correspondente a esta máquina onde se pode observar o disjuntor de proteção, a fonte de alimentação de 230V para 24VDC para a alimentação do autómato e consola. Existe ainda uma fonte de alimentação de 230V para 48VDC para alimentar o *servodrive*. Por último apresenta-se o autómato e o relé de ativação do travão magnético e ainda os bornes de ligação.



Figura 73. Quadro elétrico do divisor angular

### 5.3. Motor 60HSE3N-D25

O motor aplicado neste processo é o modelo 60HSE3N-D25, um motor de passo. As suas características são descritas na Figura 74.

Os motores de passo são dispositivos que convertem pulsos elétricos em movimentos mecânicos gerando variações angulares discretas. O motor de passo é ativo em pequenos incrementos angulares, denominados “passos”. A sequência dos pulsos que são aplicados vai diferenciar o sentido de rotação do motor (30).

| ESPECIFICAÇÕES       |                 |                         |                       | DIAGRAMA |
|----------------------|-----------------|-------------------------|-----------------------|----------|
| Âng. de Passo        | 1.8° ± 5%       | Corrente                | 5 A                   |          |
| Nº de Fase           | 2               | Resistência             | 0.45 ± 10% Ω          |          |
| Resist. de Isolação  | 100 MΩ (500VDC) | Indutância              | 1.8 ± 20% mH          |          |
| Classe de Isolamento | B               | Holding Torque          | 30kg.cm               |          |
| Ligação no driver:   | Paralelo        | A+ Vermelho<br>A- Verde | B+ Amarelo<br>B- Azul |          |

Figura 74. Especificações do motor (31)

A velocidade de rotação é dada pela frequência dos pulsos recebidos e pelo tamanho do ângulo de rotação diretamente relacionado com o número de pulsos que são aplicados. A velocidade do rotor é também diretamente afetada com o binário do motor, conforme mostra a Figura 75.

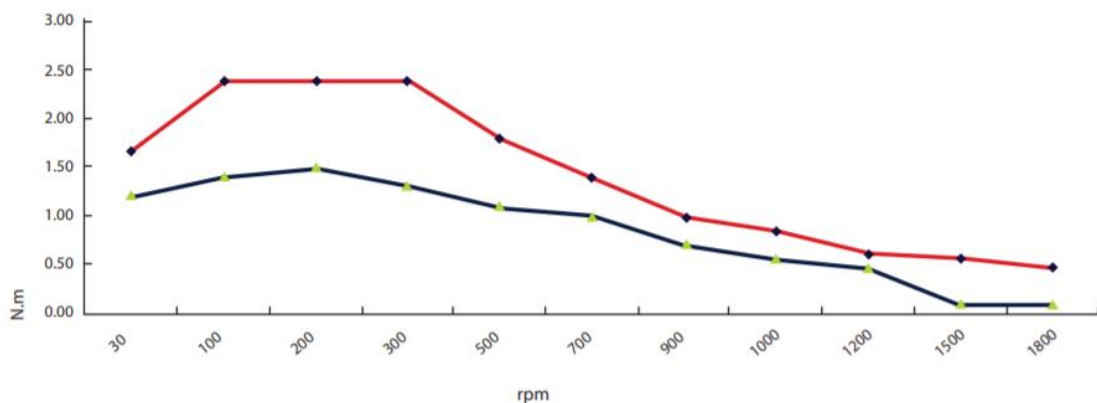


Figura 75. Curva de binário. (31)

## 5.4. Travão eletromagnético

É aplicado um travão eletromagnético no eixo da bucha (Figura 76). A função do travão é prender a bucha para que quando a CNC estiver a furar a peça o eixo não se mova.

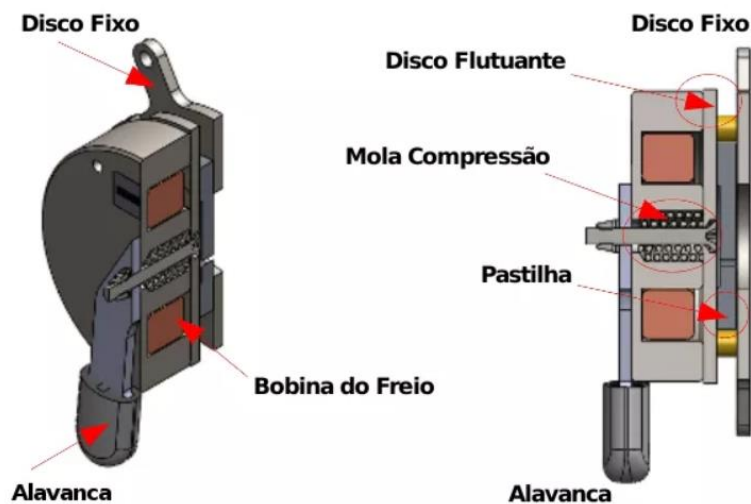


Figura 76. Constituição de um travão eletromagnético

Quando a bobine do travão é alimentada, esta gera um campo magnético que atrai o disco flutuante com mais força que a mola de compressão, puxando com maior intensidade e abrindo um pequeno espaço para que o eixo possa rodar livremente.

Sempre que a bobine não está alimentada não existe campo magnético, deixando de atrair o disco flutuante fazendo com que a mola de compressão o empurre contra a pastilha permitindo a travagem do eixo do motor.

## 5.5. Servo Drive

O *servo drive* aplicado nesta máquina é o modelo HSS60 Hybrid Drive (Figura 77). Os aspetos mais importantes neste modelo é a sua nova geração do processador de sinal digital, com 32 bits e uma tecnologia de vetores para não perder os pulsos e garantir precisão no movimento do motor. O controlo da corrente é baseado na carga, o que reduz a temperatura no motor aumentando a vida útil do mesmo.



Figura 77. Servo drive

Os *servos drives* são dispositivos de controlo de sistemas de automação, usados em muitas máquinas que requerem uma grande confiabilidade e precisão. A característica principal consiste no seu alto desempenho e alta precisão de controlo do movimento do motor devido ao facto de ser usado em malha fechada, usando um *encoder* acoplado ao motor.

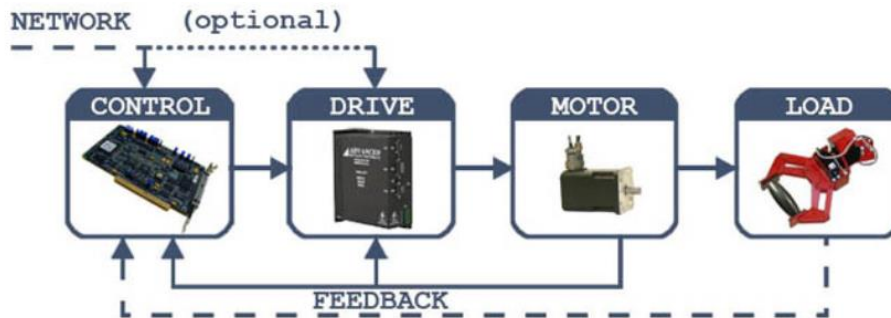


Figura 78. Malha de realimentação dos Servos Drive (32)

As entradas podem ser analógicas ou digitais para o controlo dos *servos drives*. A essência do funcionamento consiste em transformar sinais de comando de baixa para alta tensão ou corrente para ser aplicado aos enrolamentos do servo motor para produzir movimento (32).

Através deste equipamento são dados os pulsos para o motor, para que possa ajustar o ângulo introduzido pelo operador. Para uma melhor resolução do número de pulsos e para facilitar os cálculos utiliza-se 3200 pulsos por volta.

## 5.6. Autômato: CP1E

O controlo do processo é feito por um autômato, o modelo CP1E-E14SDR-A. Este modelo tem 8 entradas e 6 saídas. Em termos de comunicações este autômato permite usar a porta USB.

Na Tabela 6 encontra-se a listagem das entradas do autômato com o endereço correspondente.

Tabela 6. Entradas do autômato do Divisor Angular

| Endereço | Entradas           |
|----------|--------------------|
| COM      | COM (-V)           |
| I0.0     | Ligar/ Desligar    |
| I0.1     | <i>Start</i>       |
| I0.2     | F.C. Avançar Passo |
| I0.3     | Servo em Posição   |
| I0.4     | Alarme Servo       |
| I0.5     | Reserva            |
| I0.6     | Reserva            |
| I0.7     | Reserva            |

Na Tabela 7 encontra-se a listagem das saídas do autômato com o endereço correspondente.

Tabela 7. Saídas do autômato do Divisor Angular

| Endereço | Saídas                |
|----------|-----------------------|
| Q0.0     | Servo Pulsos Posição  |
| Q0.1     | Servo Direção CW/CCW  |
| Q0.2     | Servo Enable          |
| Q0.3     | Reserva               |
| Q0.4     | Freio da Bucha        |
| Q0.5     | Sinalizador de Alarme |

### 5.6.1. Programa de controlo

A programação do autómato é subdividida em várias secções como se pode observar na Figura 79, nomeadamente *Geral*, *Modo de Furos*, *Modo Tabela*, *Travão*, *Alarmes*, *Saídas* e *End*.



Figura 79. Secções do programa do divisor angular

#### 5.6.1.1. Geral

Esta secção é responsável por diversas operações, desde a gestão dos modos de funcionamento do processo, o *reset* aos contadores e incrementadores utilizados e a colocação da máquina em ciclo para o funcionamento.

A gestão do modo de utilização da máquina é executada através de *set's* e *reset's*. A máquina é colocada em ciclo sempre que o operador pressione o botão de *start* e sempre que o modo de utilização e de funcionamento esteja ativo.

O cálculo dos pulsos por volta é feito nesta secção:

$$\text{Pulsos de Volta} = \text{Pulsos por Volta do Servo Drive} * \text{Valor da caixa de redução}$$

O contador e incrementador são colocados a zero nesta secção. Para isso é movido o zero para o endereço de memória usado nos contadores.

As configurações da velocidade são realizadas movendo os valores das velocidades registadas no HMI para o endereço da memória usada pela função que envia os pulsos para os motores.

### 5.6.1.2. Modo de furo

O modo de furo consiste na possibilidade de o operador colocar o número de furos que deseja executar na peça a maquinar.

É efetuado o cálculo para o número de pulsos em função do valor do ângulo seleccionado pelo operador. Esse valor é movido através da função (FIXL) para o endereço de memória usado na função PLS2.

A função PLS2 envia um número especificado de pulsos para a porta seleccionada. Com esta função é definida a velocidade de aceleração, de desaceleração, a velocidade inicial e a velocidade final.

A Figura 80 mostra a especificação da porta de envio, o controlo, o endereço de memória onde são guardados os valores de aceleração e desaceleração dos pulsos e ainda um endereço de memória onde está guardada a frequência inicial de movimento.

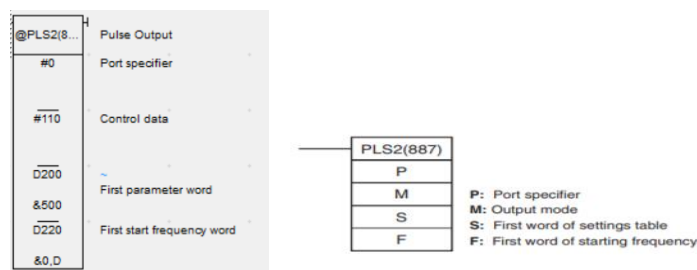


Figura 80. Exemplo da função PLS2.

A porta escolhida foi a 0, que suporta frequências até 100kHz. O conteúdo de M define os parâmetros para a saída de pulso. A escolha dos parâmetros de saída de pulso e dados de controlo seleccionou-se através da (Figura 81).

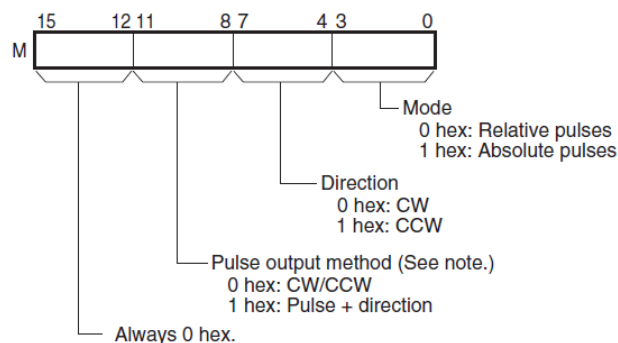


Figura 81. Configuração do pulso.

O parâmetro Mode é o modo de como se faz a leitura dos pulsos. Com *Relative pulses* os pulsos do movimento são iguais aos números de pulsos definidos. No método de *Absolute pulses* o número de pulsos do movimento é igual ao número de pulsos definidos a subtrair como o *present value* (PV). Para este processo utilizou-se o modo de *Relative pulses*, uma vez que não há necessidade de saber qual a posição anterior do motor.

O próximo parâmetro é o centro de rotação do motor. Este vai variar uma vez que é possível que o motor rode no sentido horário ou no sentido anti-horário. A solução que foi aplicada consistiu em usar a função duas vezes, nomeadamente uma em que o parâmetro aplicado fosse o sentido horário e outra no sentido anti-horário. Por fim, no parâmetro de saída de pulso aplicou-se o Pulse + direção.

### 5.6.1.3. Modo Tabela

O modo tabela tem o intuito de fazer uma pequena tabela com o número de grau e o número de repetição do grau.

Na Figura 82, abaixo representada, verificou-se o funcionamento deste método. Numa fase inicial é feito um tratamento dos dados que vem do HMI para o autómato. A partir desses dados são calculados os valores de pulsos por linha e o valor dos graus em relação às repetições. É verificado se os valores são positivos ou negativos, uma vez que a função PLS tem de ser configurada para valores negativos ou positivos.

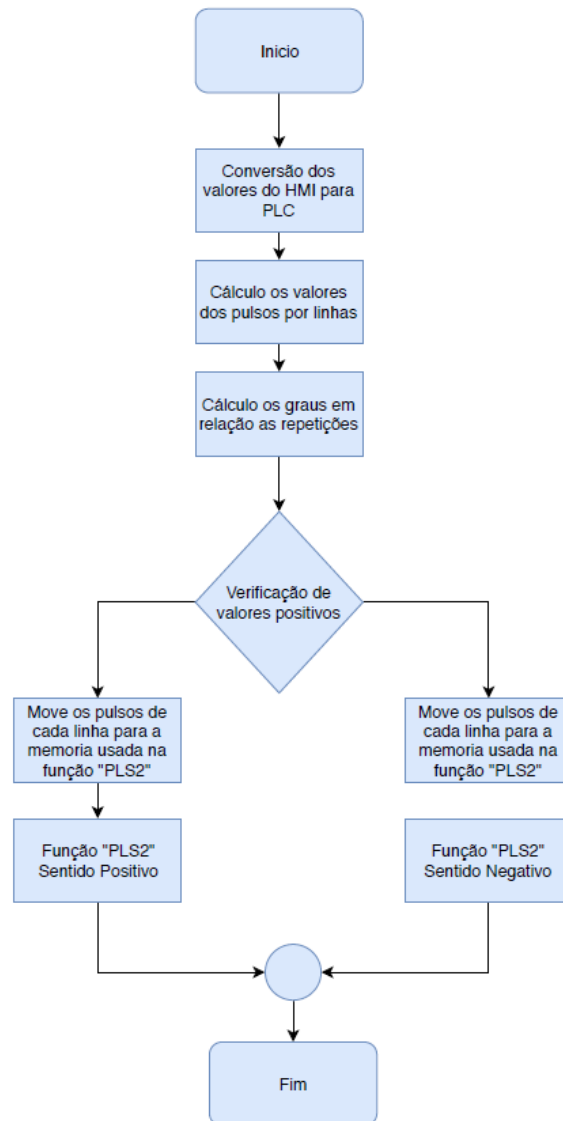


Figura 82. Fluxograma do funcionamento do modo tabela

O incremento dos valores de repetição é feito através de um fim de curso, que a ferramenta da CNC faz acionar automaticamente durante o processo. Existe ainda a possibilidade de o operador acionar manualmente sem que a máquina esteja em ciclo.

#### 5.6.1.4. Travão

Esta secção é responsável pela ativação e desativação do travão.

Na ativação é aplicado um temporizador de atraso com 5 segundos, para que o sistema tenha tempo para acabar a rotação completa do motor.

Nesta máquina existem dois modos para a ativação do travão em manual ou automático. No modo manual é ativo quando o operador selecionar na consola o botão para a sua ativação. O modo automático acontece quando o tempo de retardo estiver ativo.

O travão pode ser desabilitado de duas formas, em manual e automático. Em manual é desativado quando o operador quiser colocar a peça no ângulo zero ou quando for necessário algum ajuste no eixo.

#### 5.6.1.5. Alarmes

Neste processo os alarmes são dados pelo *servo drive*, que tem uma saída específica que é ligada a uma entrada do autómato.

Durante os ensaios foi possível observar que em qualquer movimento do motor, essa saída dava um pequeno pulso. Para resolver-se esse problema foi aplicado um filtro com um temporizador.

#### 5.6.1.6. Saídas

Esta é responsável por fazer um espelho das saídas usadas no processo com as saídas físicas do autómato. Esta técnica é usada para ser mais fácil a troca das saídas físicas (Figura 83).



Figura 83. Saídas do Divisor Angular

## 5.7. Consola: NB7W

A interface homem-máquina é feita pela consola NB7W, do modelo TW00B (Figura 84), esta tem uma dimensão de 7 polegadas com 128MB de memória interna, utilizando uma comunicação série com o interface físico do USB.



Figura 84. Consola aplicada no processo.

### 5.7.1. Sistema de supervisão

O programa utilizado para a supervisão da máquina é o NB-Designer. Este é desenvolvido pela Omron para a gama de consolas de NB.

A configuração do HMI com o autómato é feita diretamente no programa, como mostra a Figura 85.

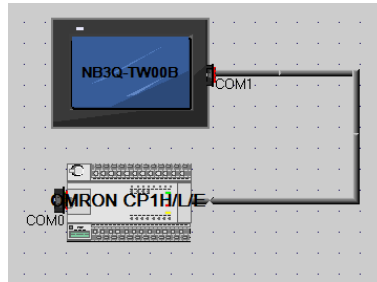


Figura 85. Configuração entre PLC e HMI no NB-Designer

A nível de programação, as memórias usadas no autómato são diretamente usadas nos HMI, como se pode observar na Figura 86.

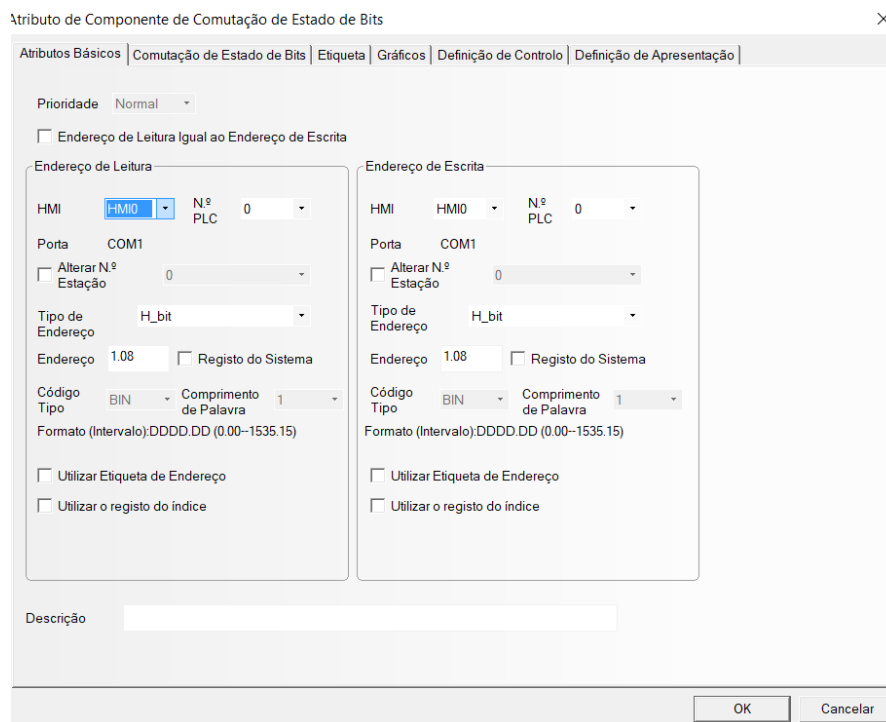


Figura 86. Atribuir um bit a um botão

A Figura 87 mostra o *template* da página principal, onde é possível ir para as páginas de modo de furos, modo de tabela de pontos, alarmes, diagnósticos e configurações da máquina.



Figura 87. *Template* da página principal

A Figura 88 mostra o modo de tabela, sendo possível observar a tabela e onde se apresentam as colunas para a rotação, repetição e a repetição atual. À direita da página observam-se diversos botões que são usados para ativar o modo de funcionamento e colocar a máquina em manual. Só é possível ativar o travão se a máquina tiver em manual. O secção onde se encontra o “Modo”, quando a máquina fica em ciclo muda de cor para vermelho. Esta só é ativa quando o modo de funcionamento é ativado.

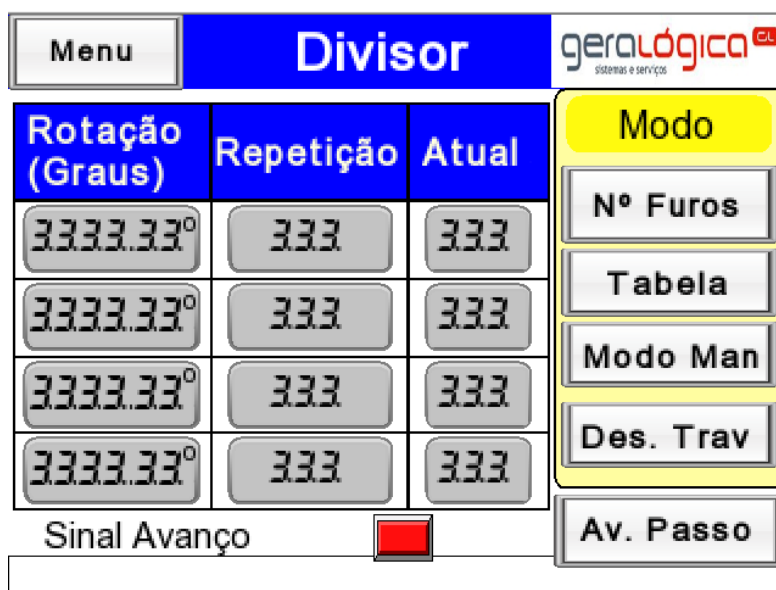


Figura 88. Modo de tabela de pontos

## 6. Conclusão

Este relatório foi elaborado de modo a apresentar uma visão do trabalho realizado durante o período em estágio, com a descrição dos projetos desenvolvidos mais relevantes.

O primeiro projeto apresentado consiste numa máquina para realizar o processo de rotomoldagem. O processo é igualmente conhecido por moldagem rotacional, este é um processo mecânico para moldagem de peças de plástico usando matéria-prima não sólida, neste caso, o polímero.

Foi desenvolvida uma máquina do tipo *Shuttle Machines*, constituída por dois carros e uma estufa, onde se encontra o queimador. Os carros são constituídos por 2 eixos, que permitem uma rotação de 360 graus do molde para uma distribuição uniforme do material. O deslocamento do molde para a seção da estufa e da zona de arrefecimento forçado é realizado através dos carros.

A estufa tem como função o aquecimento do polímero em granulado para que esta funda e se distribua de maneira homogénea pelo molde. A gestão da estufa é feita através de um sistema de supervisão. O aquecimento é realizado por cinco patamares o que permite alterar a temperatura da estufa consoante o polímero em aplicação. Nestes patamares é possível determinar a temperatura, duração do patamar e velocidade dos eixos, permitindo assim a gestão do aquecimento e arrefecimento do forno.

O sistema de supervisão foi implementado pelo programa Movicon 11. Esta solução permite aplicações de *software* de supervisão, controlo e aquisição de dados com interface de operação, onde os dados podem ser acedidos remotamente ou localmente.

O sistema de supervisão desenvolvido possui várias funcionalidades, desde a monitorização e controlo do processo em tempo real, à possibilidade da realização de receitas para diferentes moldes. Contém ainda uma base de dados com todos os moldes, produtos, matérias e ainda a classificação de cada molde. Possibilita a elaboração dos relatórios de produção, para a avaliação da produtividade da máquina, verificação e relatório de alarmes.

O projeto de supervisão foi realizado na sua totalidade. Os testes mais aprofundados e respetivas otimizações ficaram fora do âmbito deste estágio, estando da responsabilidade da empresa acolhedora.

No segundo projeto foi desenvolvido um acessório para uma CNC, que permite aumentar a capacidade de resposta adicionando um terceiro eixo a uma máquina com dois eixos, com um custo reduzido. O eixo acoplado é rotativo permitindo a rotação de peças durante o processo de maquinação de forma automática. Neste projeto foram executados os esquemas elétricos, programação do sistema de controlo e sistema de supervisão.

Este estágio demonstrou as dificuldades do mundo do trabalho pois é necessário ter um conhecimento alargado. O conhecimento tem de ir desde a criação de quadros elétricos, à eletrificação dos processos a programação de *software* e a criação dos componentes mecânicos. As principais dificuldades deste estágio foram a pressão criada pelo cumprimento de prazos e o tempo de estágio pois não pude realizar os testes do sistema de supervisão.

Por fim, o desenvolvimento deste estágio proporcionou um crescimento profissional, possibilitando um contacto com um ambiente de produção na área da automação industrial e respetivos sistemas de supervisão. No trabalho realizado foi possível verificar a complexidade da área de automação, os cuidados a ter para que um processo seja bem idealizado tendo em conta a sua fiabilidade e robustez, assim como as necessidades do cliente.

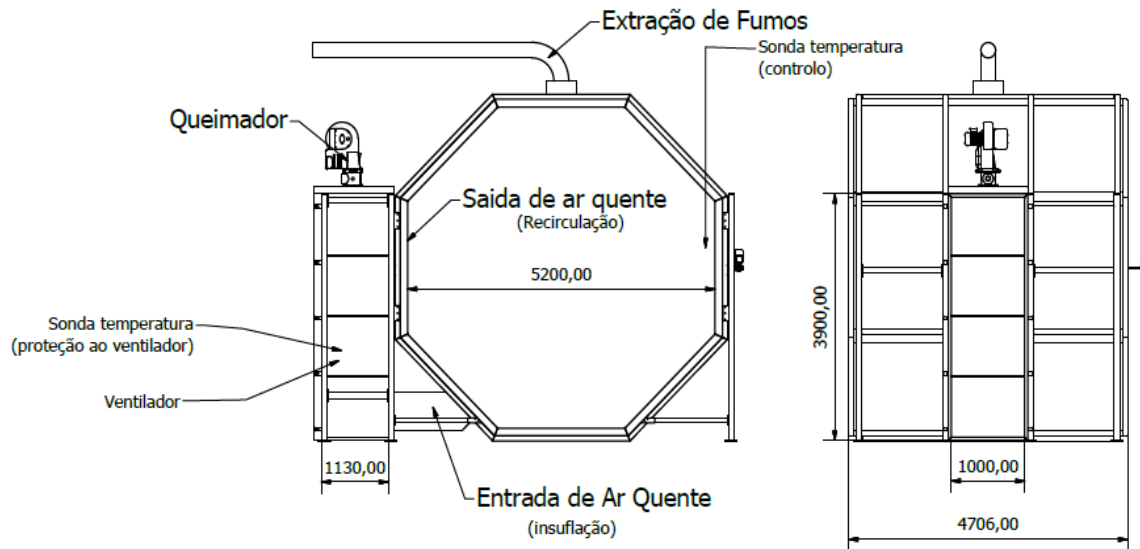
## Bibliografia

1. **Hayden, Ernie, Assante, Michael e Conway, Tim.** *An Abbreviated History of Automation and ICS Cybersecurity.* s.l. : SANS, 2014.
2. **Silevira, Leonardo e Lima, Weldson.** Um breve histórico conceitual da Automação Industrial e Redes para Automação. 2003, UFRN-PPgEE.
3. **Silva, Gustavo Monteiro da.** *AS REDES DE CAMPO EM INSTRUMENTAÇÃO.* Setúbal : Electrónica, 2004. ISBN: 972-98624-7-8.
4. **YAMAGUCHI, MARCOS YUKIO.** *Sincronização das Bases de Tempo de CLPs Distribuídos.* São Paulo : Universidade de São Paulo, 2006. Tese de Mestrado em Energia e Automação.
5. **Microsoft.** O que é o ERP e porque precisa dele? *Dynamics Microsof.* [Online] [Citação: 16 de 04 de 2019.] <https://dynamics.microsoft.com/pt-pt/erp/what-is-erp/>.
6. **Inductive Automation.** What is SCADA? [Online] 12 de 09 de 2018. [Citação: 23 de 08 de 2019.] <https://inductiveautomation.com/resources/article/what-is-scada>.
7. **Santos, Bruno P., et al.** *Internet das Coisas: da Teoria à Prática.* Belo Horizonte, MG, Brasil : Universidade Federal de Minas Gerais, 2016.
8. **SILVEIRA, Paulo R. e SANTOS, Winderson E.** *Automação e Controle Discreto.* São Paulo : Editora Érica, 2007. ISBN: 85-7194-591-8.
9. **Pinto, Pedro.** Redes – Sabe o que é o modelo OSI? [Online] Pplware, 15 de 09 de 2010. [Citação: 15 de 09 de 2019.] <https://pplware.sapo.pt/tutoriais/networking/redes-sabe-o-que-e-o-modelo-osi/>.
10. **Brotas, Marco André do Vale Pires.** *Uma Arquitetura de Transferência de Ficheiros Otimizada a Redes.* s.l. : Universidade Autónoma de Lisboa, 2013. Mestrado em Engenharia e Tecnologias Informáticas.
11. **J.Jayasamraj.** *SCADA Communication & Protocols.* Bengaluru : National Power Training Institute, 2019.
12. **National Communications System.** *Supervisory Control and Data Acquisition (SCADA) Systems.* Chantilly, Virginia : Communication Technologies, Inc., 2004.
13. **Johnson, Chris.** *Securing Safety-Critical SCADA in the Internet of Things .* Glasgow : School of Computing Science, University of Glasgow.
14. **Gohil, Hardik.** *Has SCADA had its day with the advent of IoT?* s.l. : Softwebsolutions, 2018.
15. **Lanner.** Fourth Generation SCADA Systems: Modernizing Remote Industrial Monitoring. [Online] 06 de 11 de 2017. [Citação: 2019 de 06 de 02.] <https://www.lanner-america.com/blog/fourth-generation-scada-systems-modernizing-remote-industrial-monitoring/>.

16. **Mossin, Eduardo André.** *Diagnóstico automático de redes profibus*. São Carlos : Universidade de São Paulo, 2012. Tese de Doutorado em Ciências do Programa de Engenharia elétricas.
17. **Odva.** Technology Standards. [Online] [Citação: 27 de 06 de 2019.] <https://www.odva.org/Technology-Standards/EtherNet-IP/Overview>.
18. **Costa, Paulo Alexandre Ricardo.** *Integração e supervisão de múltiplas redes de automação*. Lisboa : Instituto Superior de Engenharia de Lisboa, 2016. Tese de Mestrado em Engenharia Eletrotécnica – Ramo de Automação Industrial.
19. **Picotech.** Modbus serial protocol decoding. [Online] [Citação: 26 de 02 de 2019.] <https://www.picotech.com/library/oscilloscopes/modbus-serial-protocol-decoding>.
20. **Jorge, André Filipe Pereira.** *Sistema de Comunicação para Monitorar Consumos de Energia*. Lisboa : Universidade Nova de Lisboa, 2010. Tese de Mestrado em Engenharia Electrotécnica e de Computadores.
21. **Progea.** Movicon. [Online] 18 de 03 de 2019. <https://www.progea.com/en/movicon-11/>.
22. **Microsoft.** Administrar origens de dados ODBC. [Online] [Citação: 14 de 03 de 2019.] <https://support.office.com/pt-pt/article/administrar-origens-de-dados-odbc-b19f856b-5b9b-48c9-8b93-07484bfab5a7>.
23. —. Microsoft ActiveX Data Objects. *Microsoft*. [Online] 11 de 08 de 2018. [Citação: 16 de 06 de 2019.] <https://docs.microsoft.com/en-us/sql/ado/microsoft-activex-data-objects-ado?view=sql-server-2017>.
24. **Crawford, Roy J. e Throne, James L.** *Rotational Molding Technology*. Norwich, New York : Plastics Desing Library Willian Andrew Publishing, 2002. ISBN 1-884207-85-5.
25. **Rotoline.** Shuttle. [Online] 23 de 02 de 2019. <https://www.rotoline.com/en/rotomolding/machines/cGftlZwCX5h5nXLC5AzIDQ==>.
26. **Yaskawa Electric.** *MEMOBUS/Modbus Communications*. YASKAWA ELECTRIC : s.n., 2019.
27. **Omron.** *Digital Temperature Controllers (h175 e5c)*. s.l. : Omron, 2019.
28. **Reersafety.** Safety Light Curtains. [Online] [Citação: 21 de 03 de 2019.] <https://www.reersafety.com/us/en/products/safety-light-curtains>.
29. **Omron.** *G9SP Series, Safety Controller*. s.l. : Omron, 2010.
30. **Brites, Felipe Gonçalves e Santos, Vinicius Puga de Almeida.** *Motor de Passo*. Niterói : PET-Tele, 2008.
31. **Policomp.** *DATASHEET: 60HSE3N-D25*. São Paulo : s.n., 2010.
32. **ADVANCED Motion Controls.** What is a Servo Drive? *ADVANCED*. [Online] [Citação: 20 de 03 de 2019.] <https://www.a-m-c.com/servo-drive-top-benefits-motion-control-applications/>.

## Anexos

### Anexo 1: Dimensões da estufa



## Anexo 2: Relatórios de Produção

### Relatório de Produção



Data: 23-01-2019

Hora: 16:05:12

Filtro Data: 23-01-2019

| Número Carro | Número Ciclo | Receita        | Data Ciclo 1        | Hora Inicial Ciclo 1 | Hora Final Ciclo 1 | Hora Inicial Ciclo 2 | Hora Final Ciclo 2 | Hora Inicial Ciclo 3 | Hora Final Ciclo 3 |
|--------------|--------------|----------------|---------------------|----------------------|--------------------|----------------------|--------------------|----------------------|--------------------|
| 1            | 1            | Receita_1      | 23-01-2019          | 15:24:50             | 15:24:53           | 15:24:58             | 15:25:01           | 15:25:05             | 15:25:08           |
|              |              | <b>Molde 1</b> | <b>Molde 2</b>      | <b>Molde 3</b>       |                    | <b>Molde 4</b>       |                    | <b>Molde 5</b>       |                    |
|              |              | Molde_1        | Molde_3             |                      |                    |                      |                    |                      |                    |
|              |              | <b>Molde 6</b> | <b>Molde 7</b>      | <b>Molde 8</b>       |                    | <b>Molde 9</b>       |                    | <b>Molde 10</b>      |                    |
|              |              |                |                     |                      |                    |                      |                    |                      |                    |
|              |              | <b>Defeito</b> | <b>Tipo Defeito</b> |                      |                    |                      |                    |                      |                    |
|              |              | OK             |                     |                      |                    |                      |                    |                      |                    |

## Anexo 3: Relatórios de Alarmes

### Alarms report order by date time on



23-01-2019 16:06:41

Evaluated period: Last 60 days

Found date time: 13-12-2018 16:31:14 to 23-01-2019 16:00:00

#### quinta-feira, 13 de Dezembro de 2018

| Date time On        | Alarm  | Date time Off       | Duration        |
|---------------------|--|---------------------|-----------------|
| 13-12-2018 16:31:14 | Disparo Emergência Geral - Seguranças                        | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Falta de Fase - Seguranças                                   | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Térmico Ventilador Recirculação Forno - Forno        | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Variador Ventilador Extração Forno - Forno                   | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Disjuntor 24VDC Entradas - Alimentações              | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Disjuntor 24VDC Saídas - Alimentações                | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Disjuntor 24VDC Seguranças - Alimentações            | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Disjuntor 24VDC Freios - Alimentações                | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Disjuntor 230VAC Comandos - Alimentações             | 09-01-2019 12:01:41 | 19:30.27        |
| 13-12-2018 16:31:14 | Disparo Térmico Ventiladores Arrefecimento Carro 2 - Carro 2 | 09-01-2019 12:01:42 | 19:30.28        |
| 13-12-2018 16:31:14 | Disparo Térmico Portas Carro 2 - Carro 2                     | 09-01-2019 12:01:42 | 19:30.28        |
| 13-12-2018 16:31:14 | Variador Ventilador Recirculação Forno - Forno               | 09-01-2019 12:01:42 | 19:30.28        |
|                     |  |                     | Total: 18:05.27 |

#### segunda-feira, 17 de Dezembro de 2018

| Date time On        | Alarm   | Date time Off       | Duration       |
|---------------------|---|---------------------|----------------|
| 17-12-2018 16:02:00 | Sonda de Temperatura Danificada - Forno                   | 09-01-2019 12:01:41 | 19:59.41       |
| 17-12-2018 16:02:00 | Falha Comunicação Controlador Temperatura - Com_Forno     | 09-01-2019 12:01:41 | 19:59.41       |
| 17-12-2018 16:02:00 | Falha Comunicação Variador Translação Carro 1 - Com_Carro | 09-01-2019 12:01:42 | 19:59.42       |
| 17-12-2018 16:02:02 | Falha Comunicação Variador Rotação Eixo 1 Carro 1 - Com C | 09-01-2019 12:01:42 | 19:59.40       |
|                     |   |                     | Total: 7:58.44 |

#### segunda-feira, 21 de Janeiro de 2019

| Date time On        | Alarm                                 | Date time Off       | Duration        |
|---------------------|---------------------------------------|---------------------|-----------------|
| 21-01-2019 15:00:16 | Array Alarmes_16 - Forno              | 22-01-2019 08:54:29 | 17:54.13        |
| 21-01-2019 15:00:16 | Diferencial Temperatura Forno - Forno | 22-01-2019 08:54:29 | 17:54.13        |
|                     |                                       |                     | Total: 11:48.26 |

#### terça-feira, 22 de Janeiro de 2019

| Date time On        | Alarm   | Date time Off       | Duration        |
|---------------------|---|---------------------|-----------------|
| 22-01-2019 08:54:59 | Variador Rotação Eixo 1 Carro 2 - Carro 2             | 22-01-2019 09:00:56 | 0:05.57         |
| 22-01-2019 09:06:01 | Variador Rotação Eixo 1 Carro 2 - Carro 2             | 22-01-2019 09:06:12 | 0:00.11         |
| 22-01-2019 09:07:47 | Disparo Emergência Geral - Seguranças                 | 22-01-2019 09:29:23 | 0:21.36         |
| 22-01-2019 09:48:53 | Disparo Emergência Geral - Seguranças                 | 22-01-2019 10:01:55 | 0:13.02         |
| 22-01-2019 10:02:05 | Disparo Emergência Geral - Seguranças                 | 22-01-2019 10:02:11 | 0:00.06         |
| 22-01-2019 11:51:08 | Disparo Emergência Geral - Seguranças                 | 22-01-2019 11:51:17 | 0:00.09         |
| 22-01-2019 12:09:03 | Falha Comunicação Controlador Temperatura - Com_Forno | 22-01-2019 12:09:30 | 0:00.27         |
| 22-01-2019 14:03:04 | Time Out Avançar Carro 1 - Temporizadores             | 22-01-2019 14:03:39 | 0:00.35         |
| 22-01-2019 14:03:14 | Diferencial Temperatura Forno - Forno                 | 22-01-2019 14:03:39 | 0:00.25         |
| 22-01-2019 14:03:14 | Array Alarmes_16 - Forno                              | 22-01-2019 14:03:39 | 0:00.25         |
| 22-01-2019 14:04:40 | Time Out Recuar Carro 1 - Temporizadores              | 22-01-2019 14:05:00 | 0:00.20         |
| 22-01-2019 17:22:22 | Falha Comunicação Controlador Temperatura - Com_Forno | 23-01-2019 07:42:46 | 14:20.24        |
|                     |   |                     | Total: 15:03.37 |

#### quarta-feira, 23 de Janeiro de 2019

| Date time On        | Alarm   | Date time Off       | Duration |
|---------------------|---|---------------------|----------|
| 23-01-2019 09:45:40 | Falha Comunicação Variador Rotação Eixo 1 Carro 1 - Com_C | 23-01-2019 09:54:49 | 0:09.09  |
| 23-01-2019 11:59:59 | Diferencial Temperatura Forno - Forno                     | 23-01-2019 12:04:35 | 0:04.36  |
| 23-01-2019 11:59:59 | Array Alarmes_16 - Forno                                  | 23-01-2019 12:04:35 | 0:04.36  |