

Faculdade de Engenharia da Universidade do Porto



**Sistema de Controlo e Monitorização Remota de
uma Unidade Industrial de PSA**

Paulo Filipe Loureiro Ferreira

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Automação

Orientador: Prof. Dr. Joaquim Gabriel Magalhães Mendes

1 de Março de 2010

A Dissertação intitulada

"SISTEMA DE CONTROLO E MONITORIZAÇÃO REMOTA DE UMA UNIDADE INDUSTRIAL DE PSA"

foi aprovada em provas realizadas em 26/Fevereiro/2010

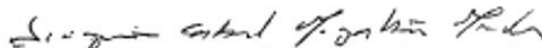
o júri



Presidente Professor Doutor Armando Jorge Miranda de Sousa
Professor Auxiliar do Departamento de Engenharia Electrotécnica e de Computadores da
Faculdade de Engenharia da Universidade do Porto

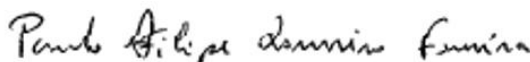


Professor Doutor Paulo Jorge Ribeiro da Cruz
Product & Business Development, da EuroResinas Industrias Químicas SA



Professor Doutor Joaquim Gabriel Magalhães Mendes
Professor Auxiliar do Departamento de Engenharia Mecânica da Faculdade de Engenharia
da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.



AUTOR: PAULO FILIPE LOUREIRO FERREIRA

Faculdade de Engenharia da Universidade do Porto

Resumo

No actual ambiente empresarial de um mercado global caracterizado por uma elevada concorrência e mudanças constantes, a única forma das empresas obterem uma diferenciação positiva é apostando na inovação contínua e desenvolvimento.

É neste contexto que surge este projecto, resultado de uma parceria entre a empresa Sysadvance S.A. e a Faculdade de Engenharia da Universidade do Porto, com o objectivo de criar um sistema combinado de monitorização e controlo de uma unidade industrial de PSA.

Neste projecto foi desenvolvido e testado com sucesso um sistema de monitorização e controlo baseado num PLC da OMRON. Este sistema controla os ciclos combinados de uma unidade de produção industrial de oxigénio de elevado grau de pureza. A sua operação pode fazer-se quer localmente através de uma HMI, quer remotamente usando uma aplicação desenvolvida em Visual Basic que comunica com o autómato através de um componente OCX da OMRON com o nome CX-Server Lite, possibilitando dessa forma uma parametrização fácil e intuitiva de todo o processo por parte do utilizador.

Este documento demonstra como o uso das técnicas de projecto de engenharia na concepção e desenvolvimento de um novo sistema é uma forma de garantir boas práticas e formalizar o processo de pensamento.

Demonstra-se também a importância do uso de *standards* na programação de autómatos, mais especificamente a norma IEC 61131-3 e as linguagens de programação especificadas nessa mesma norma, dando especial ênfase ao uso de *Sequential Function Charts* (SFC). Este documento torna-se, desta forma, um caso de estudo de como ferramentas de engenharia são aplicadas no desenvolvimento de novos produtos a nível industrial

Abstract

In the current business environment of high levels of competition and constant changes, the only way for enterprises to achieve differentiation in a saturated market is betting on continuous innovation and development.

In this context a project arises as a result of a partnership between the company SysAdvance S.A. and the Faculty of Engineering, University of Porto, with the aim of creating a combined system of monitoring and control of an industrial PSA unit.

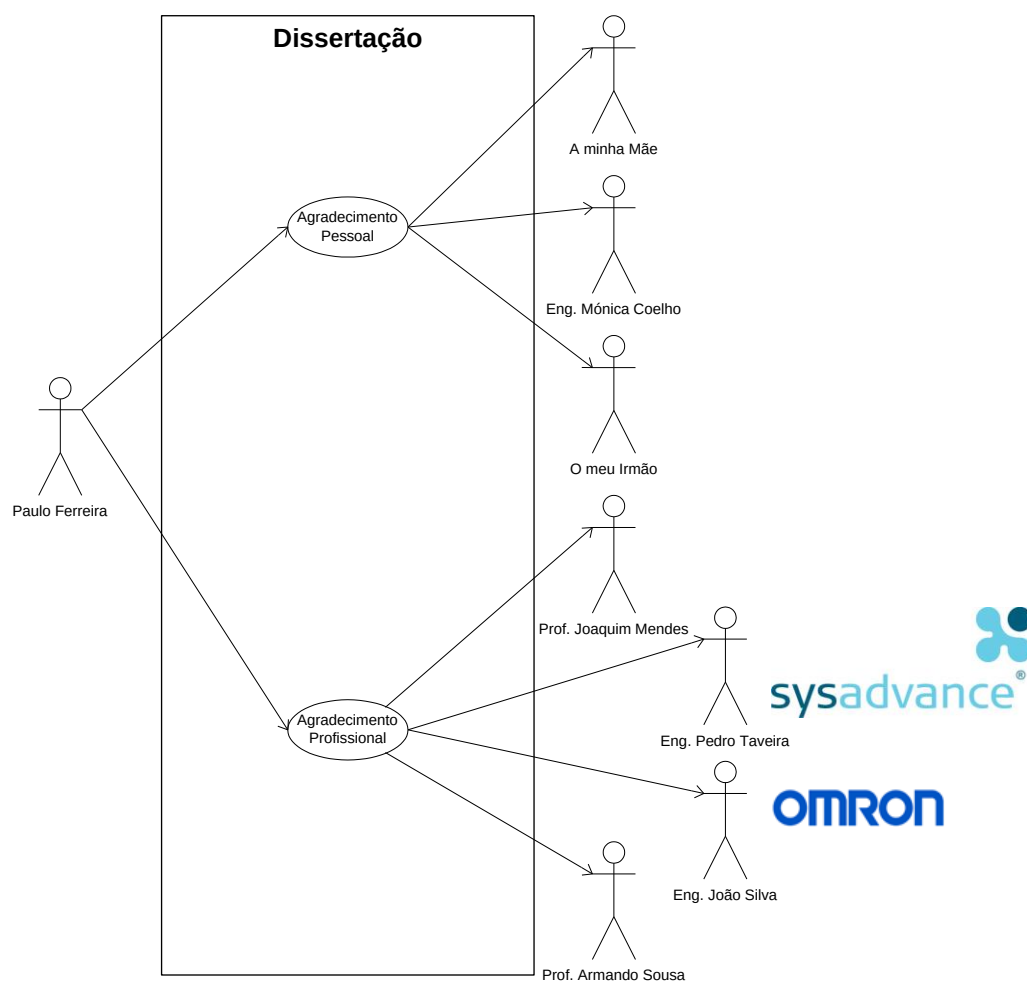
This document shows how the use of an engineering design process to create a new system is one way to ensure good practices and formalize the thought process.

It also clearly displays the importance of using standards in the programming of PLCs, more specifically the IEC 61131-3 and the programming languages specified in that provision, with particular emphasis on the use of Sequential Function Charts.

This document becomes thus a case study of how engineering tools are applied in the development of new products at industrial level.

An OMRON PLC based system for monitoring and controlling was developed and tested successfully. This system controls the combined cycles of a high purity oxygen production unit. Its operation can be done either locally via a HMI or remotely using an application developed in Visual Basic that communicates with the PLC via an OCX component called OMRON CX-Server Lite, thus enabling an easy and intuitive parameterization of the whole process by the user.

Agradecimentos



“The first rule of any technology used in a business is that automation applied to an efficient operation will magnify the efficiency. The second is that automation applied to an inefficient operation will magnify the inefficiency.”

Bill Gates

Índice

Resumo	i
Abstract.....	iii
Agradecimentos	v
Índice.....	ix
Lista de figuras	xi
Lista de tabelas	xiii
Abreviaturas e Símbolos	xiv
1 - Introdução	1
1.1. Motivação	1
1.1.1. SysAdvance.....	1
1.1.2. Identificação do problema	2
1.2. Estrutura do relatório	2
2 - Estado da arte	5
2.1. Separação dos gases atmosféricos.....	5
2.1.1. Separação por membranas	5
2.1.2. Destilação criogénica	6
2.1.3. Electrólise	6
2.1.4. Pressure Swing Adsorption	7
2.2. Soluções de controlo industrial	9
2.2.1. Autómatos programáveis	9
2.2.2. Microcontrolador	11
2.2.3. Computadores industriais	11
2.2.4. Soft-PLC	12
3 - Especificação dos Requisitos	13
3.1. Identificação de Necessidades	13
3.1.1. Recolha das informações gerais	14
3.1.2. Organização hierárquica.....	14
3.1.3. Determinação da importância	15
3.2. Requisitos Técnicos	16

4 - Solução Tecnológica	17
4.1. Arquitectura do sistema	17
4.2. Design de sistema	17
4.2.1. Decomposição Funcional	17
4.2.2. Modelos de comportamento	21
5 - Protótipo	25
5.1. Hardware	25
5.1.1. PLC	25
5.1.2. Comunicação GPRS	26
5.1.3. Human-Machine Interface (HMI)	27
5.1.4. Sistema de controlo	28
5.2. Software	30
5.2.1. Software usado	30
5.2.2. Programação	31
6 - Conclusão	39
6.1. Comentários finais	39
6.2. Futuros desenvolvimentos	40
Hardware	40
Software	41
Referências	43

Lista de figuras

Figura 1 - Diagrama das fases do processo de projecto de engenharia e os capítulos correspondentes.....	3
Figura 2 - Esquema simplificado do processo da electrólise da água.....	7
Figura 3 - Unidade PSA de produção de azoto de elevado débito (esquerda); PSA de débito médio (direita).....	7
Figura 4 - Diagrama simplificado de um processo de PSA genérico.....	8
Figura 5 - Arquitectura de um autómato modular (catálogo Siemens).....	10
Figura 6 - Fontes de informação para a especificação dos requisitos segundo o IEEE Std. 1233-1998.....	13
Figura 7 - Diagrama hierárquico de necessidades do cliente	15
Figura 8 - Diagrama funcional de alto nível do sistema de controlo - vista do hardware.....	18
Figura 9 - Diagrama funcional de baixo nível do sistema de controlo - vista do hardware	19
Figura 10 - Diagrama funcional de baixo nível do software de controlo	20
Figura 11 - Diagrama de estados UML do bloco selecção de modo.....	22
Figura 12 - Diagrama de estados UML do bloco controlo de operação	23
Figura 13 - Diagrama de estados UML de um bloco de controlo de ciclo com 12 etapas genérico.....	24
Figura 14 - PLC Omron CJ1M CPU11	25
Figura 15 - Modem Moxa OnCell G2110T.....	27
Figura 16 - Consola HMI OMRON NS5	27
Figura 17 - Montagem do sistema de controlo no armário eléctrico (esquerda); pressostato digital e válvula (direita)	28
Figura 18 - Vista do armário do sistema de controlo com porta fechada	29
Figura 19 - Unidade PSA experimental e sistema de controlo	29

Figura 20 - Software CX Programmer da Omron.....	30
Figura 21 - Software CX Designer da Omron	31
Figura 22 - Componentes CX-Server Lite da Omron em ambiente Excel+VBA	31
Figura 23 - Comparação entre linguagem de programação SFC e diagrama de estados UML ..	32
Figura 24 - Vista do código de controlo principal em SFC	32
Figura 25 - Lista de variáveis usadas na parametrização de um dos macro-estados	34
Figura 26 - Aplicação de parametrização remota em Microsoft Excel	35
Figura 27- Sinóptico de apresentação.....	35
Figura 28 - Sinóptico do menu principal	36
Figura 29 - Sinóptico de manutenção	36
Figura 30 - Sinóptico de parametrização	37
Figura 31 - Sinóptico de gráficos.....	37
Figura 32 - Sinóptico de alarmes.....	37

Lista de tabelas

Tabela 1 - Arquitectura de alto nível do sistema	17
Tabela 2 - Entradas e saídas digitais do PLC Omron	26
Tabela 3 - Entradas e saídas analógicas do PLC Omron	26

Abreviaturas e Símbolos

Lista de abreviaturas (ordenadas por ordem alfabética)

AI	Analogue input (Entradas analógicas)
CPU	Central Processing Unit
DI	Digital Input (Entradas digitais)
DIN	Deutsches Institut für Normung, Instituto alemão de standards
DO	Digital Output, Saídas digitais
GPRS	General Packet radio service
GSM	Global System for Mobile Communications
HMI	Human-Machine Interface
HTTP	Hypertext Transfer Protocol
I/O	Input/Output, Entradas e saídas
IEC	International Electrotechnical Commission (em português: CEI)
IEEE	Institute of Electrical and Electronics Engineers
LD	Ladder, Linguagem de programação descrita na norma IEC 61131-3
PLC	Programmable Logic Controller
PSA	Pressure Swing Adsorption
RTD	Resistance Temperature Detector
SFC	Sequential Function Chart
ST	Structured Text
UML	Unified Modeling Language
VBA	Visual Basic for Applications

Introdução

1.1. Motivação

Nos dias actuais, caracterizados por um ambiente de elevada competitividade empresarial e cuja concorrência é estimulada por um mundo sem fronteiras, aliado a uma enorme velocidade de comunicação, a inovação de um produto é cada vez mais uma necessidade das empresas para a conquista ou manutenção da sua quota do mercado.

A inovação é um processo de criação de valor acrescentado em contexto de mudança e como tal torna-se um factor diferenciador num meio extremamente competitivo.

O desenvolvimento de produtos inovadores e/ou a evolução de produtos existentes é um esforço empresarial complexo que envolve a integração de diversas competências e planeamento das várias fases do projecto. É pois necessária uma visão integradora de todo o projecto, desde a identificação de necessidades do mercado e a geração de conceitos inovadores, desenvolvimento e produção até ao produto que satisfaça essas mesmas necessidades.

É neste contexto que surge este projecto, resultado de uma parceria entre a empresa SysAdvance S.A. e a Faculdade de Engenharia da Universidade do Porto, com o objectivo de criar um produto inovador na área de separação de gases industriais.

1.1.1. SysAdvance

A SysAdvance é uma empresa sediada na Maia, especializada na separação de gases industriais. Com uma forte componente de investigação e desenvolvimento, reforçada pelas parcerias com Laboratórios de Universidades, providencia soluções de elevado valor acrescentado a nível industrial e médico.

O elevado grau de especialização em tecnologias de separação de gases, como o PSA ou membrana, permite-lhes oferecer soluções personalizadas aos clientes.

No resto deste documento a empresa SysAdvance S.A. será referida como *o cliente*.

1.1.2. Identificação do problema

Nesta subsecção coloca-se um conjunto de questões que permitem identificar e estruturar o problema a abordar neste projecto:

O que se está a tentar fazer?

O cliente está a desenvolver um processo de separação de oxigénio do ar que é uma variante de um processo de PSA simples. O objectivo é construir um sistema que controle e monitorize esse processo de forma autónoma.

Como é feito hoje e quais as limitações do processo?

Os processos de PSA são controlados por autómatos simples com pouca ou nenhuma sensorização e sem qualquer capacidade de parametrização que não envolva reprogramação do próprio autómato.

Em que medida é que o projecto inova?

Este projecto inova na quantidade de variáveis de processo que são monitorizadas, na complexidade acrescida do processo e na capacidade de parametrização do controlo de forma local e remota.

Se o projecto for bem sucedido que diferença irá fazer?

Com a extensão da sensorização, o registo temporal de todas as variáveis do processo e a sua parametrização remota, o cliente terá a possibilidade de executar um conjunto de testes ao processo de PSA que até à altura não tem meios para fazer.

Quanto tempo irá durar o projecto?

O projecto será realizado simultaneamente com a dissertação e como tal terá a duração de um semestre.

1.2. Estrutura do relatório

Com o objectivo de formalizar o processo de pensamento de modo a garantir que são seguidas boas práticas na concepção de um sistema que solucione o problema identificado, foram usadas técnicas de acompanhamento de um projecto de engenharia que dividem o projecto em várias etapas, conforme mostrado na Figura 1.

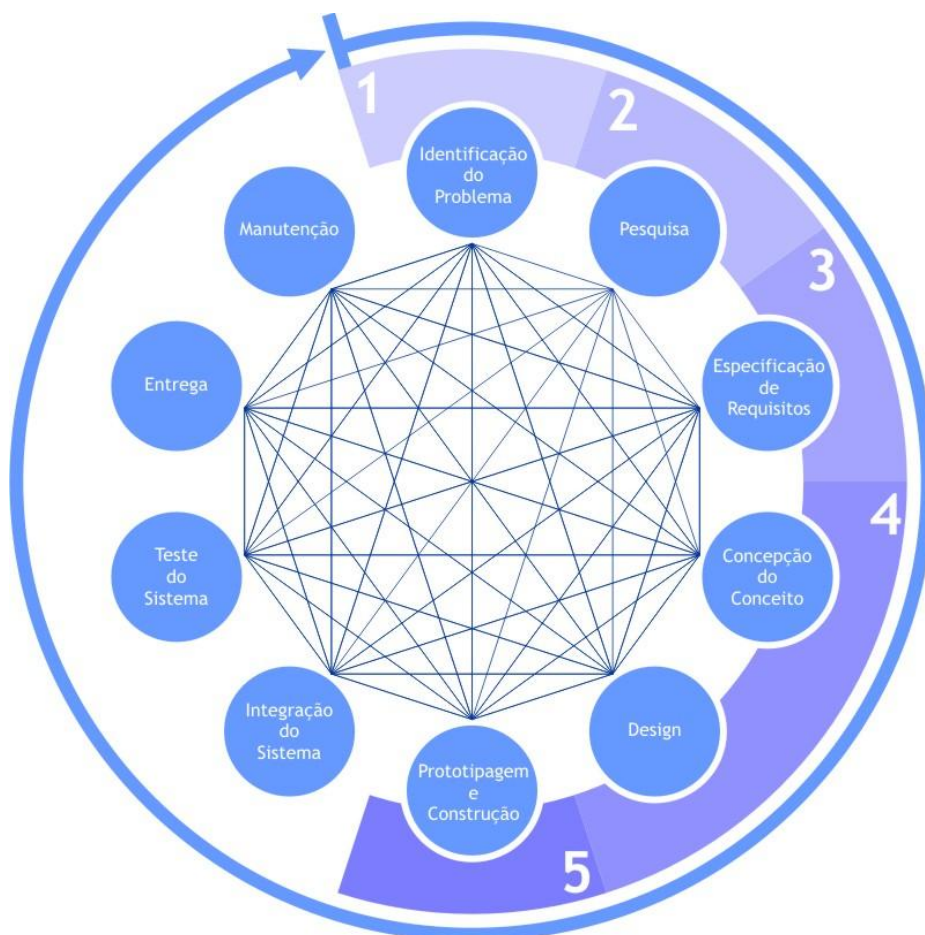


Figura 1 - Diagrama das fases do processo de projecto de engenharia e os capítulos correspondentes.

Este documento partilha da mesma estrutura do processo de projecto estando os capítulos correspondentes a cada etapa indicados na Figura 1.

Segue-se uma breve descrição de cada capítulo.

O objectivo do Capítulo 1 é descrever sucintamente e tipificar o problema do cliente, aproveitando a informação fornecida pelo cliente para responder a um conjunto de perguntas padrão que identificam se o mesmo é concretizável e/ou cria algo de inovador.

No capítulo 2 é realizada uma pesquisa de tecnologias existentes no mercado que poderão solucionar o problema do cliente. Essa pesquisa encontra-se dividida em duas partes, a primeira relativa aos principais processos de separação de gases existentes e a segunda às tecnologias que podem ser empregues para o seu controlo automático.

No capítulo 3 apresentam-se as especificações de requisitos do sistema. O capítulo começa com a identificação e análise das necessidades e requisitos funcionais do cliente que motivam a criação deste projecto. Posteriormente é feita a especificação dos requisitos técnicos e restrições provenientes do ambiente.

No capítulo 4 é apresentada a arquitectura da solução tecnológica escolhida dentro do conjunto de soluções que preenchem os requisitos do projecto. O design escolhido é decomposto em blocos funcionais com uma vista de software e uma vista de hardware e diversos níveis de detalhe funcional. De seguida, o seu comportamento é modelizado com linguagem UML.

O protótipo e aspectos importantes da sua construção são apresentados no capítulo 5. Este capítulo apresenta o hardware utilizado, seguido dos ambientes de desenvolvimento. Por fim é apresentado o software de controlo do autómato e uma pequena demonstração da aplicação criada para a HMI local e remota.

Por fim, o capítulo 6 resume as conclusões retiradas no final da dissertação e apresenta algumas ideias para implementações e futuros desenvolvimentos deste projecto.

Capítulo 2

Estado da arte

2.1. Separação dos gases atmosféricos

A produção industrial do azoto e oxigénio é efectuada quase exclusivamente a partir do ar atmosférico. Entre as tecnologias de separação para a produção destes gases, destaca-se a separação por membranas, a destilação criogénica e a adsorção por modulação de pressão (*Pressure Swing Adsorption*).

Estas tecnologias de separação baseiam-se em diferentes princípios e são caracterizadas por diferentes variáveis de processo, investimentos e custos de operação.

O oxigénio pode também ser produzido não por separação de gases a partir do ar, mas por uma quarta técnica designada de electrólise da água. Esta técnica não é contudo utilizada frequentemente na indústria dada a sua baixa eficiência.

Uma breve descrição de cada processo, com mais ênfase no processo de PSA, é apresentada de seguida.

2.1.1. Separação por membranas

A separação por membranas baseia-se nas diferentes capacidades dos gases em permear através de uma membrana, essa propriedade é designada de permeabilidade. Esta depende não só do gás mas também do tipo de membrana (geralmente polimérica). O modelo solução-difusão é muitas vezes utilizado para descrever o mecanismo de permeação - a superfície da membrana adsorve o gás no lado de alta pressão, este processo é seguido pela difusão pela membrana para o lado de baixa pressão seguido de desorção.

Através da aplicação desta tecnologia obtém-se ar enriquecido com 30 a 45% de pureza de oxigénio e como tal este processo normalmente não é usado para a produção de oxigénio de elevada pureza.

2.1.2. Destilação criogénica

A destilação criogénica é o processo mais usado para produção de oxigénio, azoto e argon a partir do ar. Mais de 90% da produção mundial é obtida através deste processo.

A destilação criogénica ou destilação fraccionada usa a diferença entre os pontos de ebulição do oxigénio, azoto e árgon para separar e purificar esses gases. Este processo é usado quando se pretende obter elevadas purezas (>99%), elevado volume de produção e produtos em estado líquido.

2.1.3. Electrólise

A electrólise da água consiste na decomposição da água em oxigénio e hidrogénio por efeito da passagem de uma corrente eléctrica pela água. A passagem de corrente eléctrica através da solução aquosa promove a decomposição da água em oxigénio e hidrogénio gasosos, obtendo-se em condições ideais duas partes de hidrogénio para cada parte de oxigénio.

Uma fonte de tensão é ligada a dois eléctrodos colocados numa solução aquosa, água com adição de um electrólito, como esquematizado na Figura 2. A fonte de alimentação, tem como função retirar electrões do eléctrodo positivo e dirigi-los para o eléctrodo negativo.

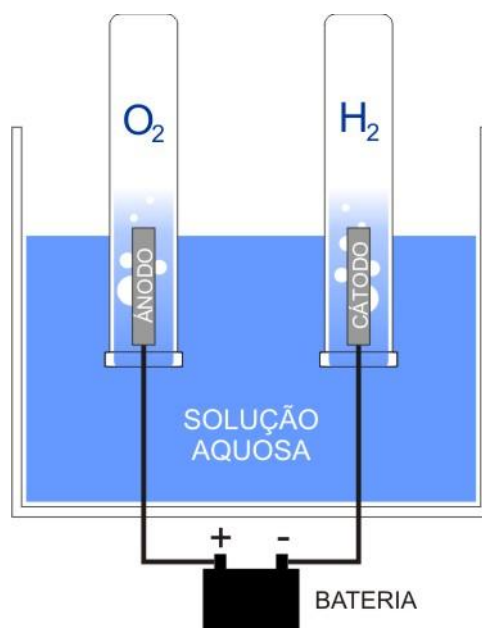


Figura 2 - Esquema simplificado do processo da electrólise da água

Deste modo ocorre uma reacção de redução no eléctrodo negativo utilizando os electrões fornecidos pela fonte. Os electrões que entram na fonte no seu terminal positivo são obtidos a partir de uma reacção de oxidação no vaso de electrólise.

2.1.4. Pressure Swing Adsorption

O processo de Pressure Swing Adsorption (PSA) é um processo de separação de ar não criogénico aplicado normalmente para obter azoto ou oxigénio no estado gasoso. Pode-se ver o aspecto de uma unidade de PSA na Figura 3.

Este processo é uma escolha rentável para produção directamente no local de utilização final e quando o débito necessário não excede algumas dezenas de toneladas por dia e a pureza de produto final pretendida é inferior a 95%.



Figura 3 - Unidade PSA de produção de azoto de elevado débito (esquerda); PSA de débito médio (direita)

A separação por PSA baseia-se na capacidade de certos materiais adsorverem diferentes gases. Esta capacidade de adsorção depende, entre outros factores, do adsorvente e do gás que é adsorvido.

No caso específico da produção de oxigénio a partir do ar atmosférico, o adsorvente utilizado costuma ser um zeólito, estando referenciados na literatura a utilização de diferentes tipos de zeólitos: zeólito 5A, zeólito 13X, zeólito LiX, zeólito LiLSX. Estes materiais

são mais ou menos sensíveis à presença de humidade no ar, pelo que esta deve ser previamente removida.

O PSA é um processo cíclico que consiste em fazer passar ar comprimido através de uma coluna cheia de um material adsorvente retendo deste modo componentes do ar não desejados enquanto que a parte restante prossegue através do meio até à saída da coluna.

Com o objectivo de regenerar o adsorvente, é necessário uma desorção periódica do leito de adsorvente. Consequentemente os sistemas de PSA têm usualmente duas colunas de adsorvente para proporcionar a continuidade operacional. Enquanto um dos recipientes produz o gás pretendido por meio de adsorção dos componentes não desejados do ar, o outro encontra-se a regenerar depressurizando para a atmosfera, Figura 4.

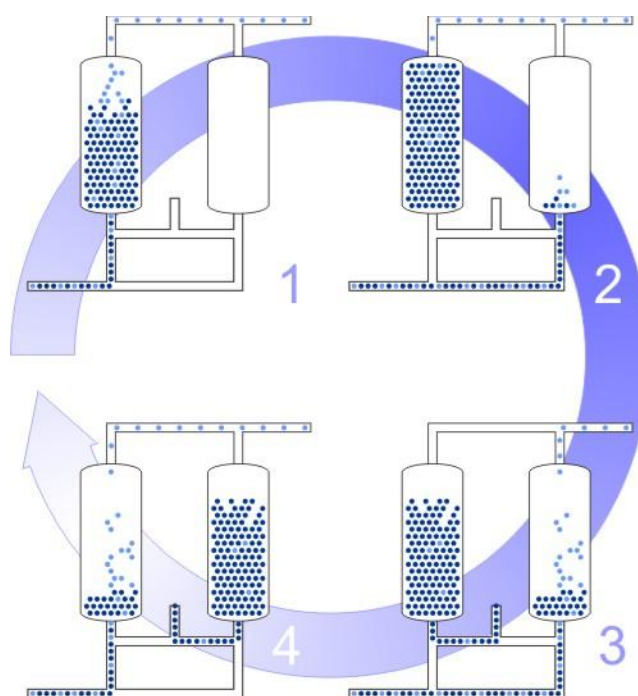


Figura 4 - Diagrama simplificado de um processo de PSA genérico

Na Figura 4 encontra-se esquematizado um processo simplificado de produção de oxigénio por PSA, onde se representam as duas fases principais, produção e regeneração, em quatro etapas diferentes. Na primeira etapa o ar comprimido é inserido no primeiro leito de adsorvente, sendo o azoto preferencialmente adsorvido e permitida passagem livre ao oxigénio. Quando o primeiro recipiente fica saturado, o fluxo de ar comprimido é direccionado para o segundo leito. Enquanto o segundo leito separa o oxigénio do azoto, o primeiro leito é despressurizado para a atmosfera, libertando o azoto que se encontrava adsorvido. No quarto instante o ciclo recomeça com o primeiro leito, garantindo assim uma produção contínua de oxigénio.

Para além das etapas de produção e regeneração, existem outras fases como a igualização e a pressurização com ar ou produto, que trazem benefícios ao processo, apesar de aumentarem a complexidade do controlo.

2.2. Soluções de controlo industrial

2.2.1. Autómatos programáveis

Os autómatos programáveis (em inglês, programmable logic controllers -PLCs-), são pequenos computadores orientados para a realização de tarefas de automação em ambiente industrial. São caracterizados por possuírem um elevado número de entradas e saídas (analógicas e digitais), capacidade de interligação a redes de comunicação de dados (profibus, devicenet, ethernet...), uso de linguagens de programação específicas, robustez de desempenho e de resistência a ambientes com características agressivas (característica que os torna ideais para ambientes industriais).

As funções de um autómato são a aquisição de dados do processo, análise dos dados, envio de comandos para o processo, troca de dados com a interface homem-máquina e gestão de comunicações com outros equipamentos.

Os autómatos podem ser compactos ou modulares. Os autómatos compactos são caracterizados por apresentarem uma construção monolítica com um número fixo de entradas e saídas, capacidade de interligação com redes de baixo custo, geralmente sem interface homem máquina, baixo custo, serem monoprocessador (capacidade de processamento limitada). Alguns exemplos deste tipo de autómatos: Schneider Micro, Omron CQM1, Siemens S200.

Os autómatos modulares são caracterizados por apresentarem uma construção modular que confere uma maior adaptabilidade às características da aplicação, serem mono ou multiprocessador (capacidade de processamento elevada) que define o número e tipo de entradas e saídas, grande variedade de cartas de entrada/saída, interface com diversos tipos de redes industriais, custo médio ou alto, robustez, interface homem máquina simples. Alguns exemplos deste tipo de autómatos: Schneider Premium, Omron CJ1, Siemens S300. Na Figura 5 ilustra-se um exemplo geral da arquitectura de hardware de um autómato tendo por base os autómatos Premium da Schneider.

Os elementos que compõem a arquitectura representada na Figura 5 são:

- Bastidor e extensões do bastidor;
- Fonte de alimentação;
- Processador;
- Módulos de comunicações;
- Módulos de entradas/saídas (analógicas e digitais);
- Módulos especiais.

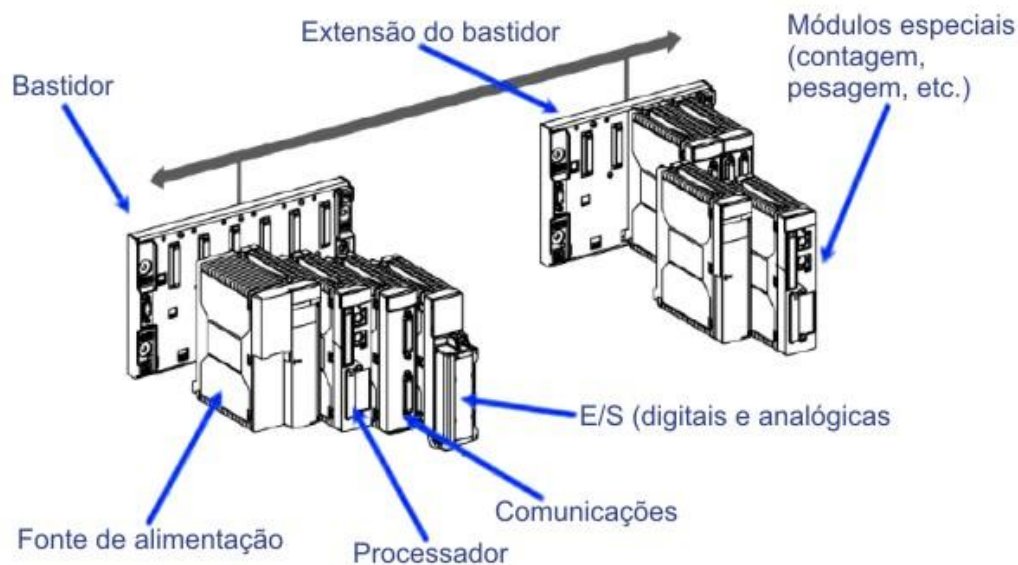


Figura 5 - Arquitectura de um autômato modular (catálogo Siemens)

Bastidores ou racks são utilizados na montagem dos vários módulos do autômato, bem como na sua fixação a uma estrutura, possuindo barramentos de controlo e comunicação entre módulos.

A fonte de alimentação fornece a alimentação aos módulos do autômato através do barramento do bastidor.

O processador recebe, via barramento do bastidor, dados provenientes dos módulos de entrada, processando essa informação através de um programa e envia, via barramento do bastidor, os dados processados para os módulos de saída.

Os módulos de comunicações permitem ao autômato trocar informações com outros dispositivos como por exemplo impressoras, outros autômatos, scadas, leitores de códigos de barras, PCs, robots, variadores de velocidade, módulos de entrada/saída, HMI, etc. As interfaces mais usadas são RS-232, RS-485 e Ethernet e os protocolos mais usados são modbus, profibus, DeviceNet e TCP/IP.

Os módulos de entradas/saídas podem ser analógicos ou digitais. São a interface entre o autômato e a parte operativa, sendo responsáveis pela conversão de sinais, com um número

de canais variável, de entradas em tensão, corrente, termopares e RTD, várias gamas de sinais de entrada, com funções de processamento, filtragem de sinais de entrada, saídas em tensão, corrente, saídas em transístor, triac ou relé.

Os módulos especiais disponibilizam à aplicação (no processador) funções pré-definidas e configuráveis com o objectivo de diminuir o tempo de execução de funções muito complexas e demoradas, como por exemplo: contagem, pesagem, leitura de codificadores, interface homem máquina, controlo de motores, lógica difusa e controlo de processos.

Para além destes tipos de módulos, existem ainda módulos de ventilação usados para introduzir ventilação forçada em situações em que seja necessário um acréscimo de ventilação e módulos de entradas/saídas distribuídos com características análogas aos módulos de entradas/saídas que permitem deslocalizar os módulos de interface com os módulos de entradas/saídas do autómato para uma localização mais próxima dos sensores e actuadores, reduzindo assim o custo da cablagem.

2.2.2. Microcontrolador

O microcontrolador consiste num circuito integrado com microprocessador, relógio, portas de entrada/saída, memória de programa, memória RAM e *timers*. Os microcontroladores são concebidos para a realização de pequenas aplicações dedicadas, geralmente sistemas de baixa complexidade, baixo custo e compactos. São fiáveis mas são igualmente pouco robustos em relação às condições ambientais. Existe uma enorme variedade de microcontroladores com diferentes características de forma a responder a diferentes requisitos que a aplicação ou sistema, possam exigir.

2.2.3. Computadores industriais

Computadores industriais são computadores adaptados ao ambiente industrial. O uso desta solução apresenta vantagens, nomeadamente:

- Uma arquitectura de hardware aberta com uma grande variedade de componentes de diversos fabricantes, o que possibilita a escolha da melhor arquitectura que se adequa ao desempenho pretendido;
- Maior capacidade de memória e processamento relativamente aos autómatos programáveis;
- Uma grande variedade de software disponível;
- Possibilidade de ter uma interface homem máquina (HMI) integrada;

- Facilidade de interligação com outros equipamentos, custos de hardware e software geralmente mais baixos em arquitecturas de média/elevada complexidade comparando com os autómatos programáveis (nomeadamente quando se pretende realizar um upgrade).

Apresenta também desvantagens como:

- Custos elevados para arquitecturas de baixa complexidade;
- Necessidade de manutenção do sistema operativo e das aplicações;
- Suporte técnico inferior ao obtido para os autómatos programáveis;
- Mais caro e menos robusto que os PLC's;
- Necessidade de cartas adicionais para fazer a interface com o processo produtivo.

2.2.4. Soft-PLC

Soft-PLC é basicamente software executado num computador pessoal que implementa um funcionamento similar a um autómato programável, suportando ambientes de desenvolvimento baseados no IEC 61131, ou seja, um PLC virtual. Esta implementação necessita de um sistema operativo que garanta a execução do programa em tempo real.

Capítulo 3

Especificação dos Requisitos

3.1. Identificação de Necessidades

Segundo o *IEEE Guide For Developing System Requirements Specifications (Std. 1233-1998)*, existem três fontes de entrada de informação para o desenvolvimento de uma especificação de requisitos, como mostrado na Figura 6.



Figura 6 - Fontes de informação para a especificação dos requisitos segundo o IEEE Std. 1233-1998

Do cliente surgem as necessidades ou requisitos funcionais, da comunidade técnica os requisitos de engenharia e do ambiente obtém-se os *standards* e restrições aplicáveis ao projecto.

Começaremos com a identificação das necessidades do cliente.

3.1.1. Recolha das informações gerais

Após algumas sessões de *brainstorming* para a recolha das necessidades do cliente foram identificadas as seguintes situações.

O cliente possui actualmente sistemas de produção de oxigénio através da tecnologia de PSA de baixa complexidade com pouca sensorização, controlados por autómatos de baixa gama. Estes autómatos correm algoritmos simples e pouco ou nada parametrizáveis, o que implica que qualquer reconfiguração, mesmo de baixa dificuldade, tenha de ser realizada por alguém especializado em programação de autómatos.

O cliente pretende criar um produto novo, evolução do anterior, cujo processo é bastante mais complexo. Este deverá ser capaz de controlar e implementar várias situações de funcionamento e reagir a ordens de emergência / stop de uma forma controlada sem colocar em perigo as pessoas ou os bens materiais. Deverá ainda ser capaz de adquirir em tempo real as diversas variáveis de processo e registar periodicamente os seus valores num sistema de armazenamento de onde se possa descarregar posteriormente esse registo para visualização e tratamento estatístico.

É pretendido que o novo produto seja totalmente reconfigurável e monitorizável por qualquer tipo de utilizador, independentemente do seu grau de conhecimento de linguagens de programação. Deverá por isso ser bastante intuitivo e simples. O novo sistema terá de ter dimensões reduzidas, baixo consumo e resistir a ambientes agressivos.

O sistema terá de ser facilmente monitorizável e parametrizável local e remotamente, sendo que a comunicação remota com o sistema deverá ser bastante versátil baseada numa rede móvel de modo a limitar o menos possível a sua utilização.

3.1.2. Organização hierárquica

A partir das informações gerais do subcapítulo anterior, retirou-se um conjunto de frases sintéticas e claras que clarificam as necessidades do cliente. Posteriormente organizaram-se essas necessidades de forma hierárquica desde o grau mais geral até ao grau mais detalhado agrupando as necessidades que possuem semelhança funcional. Na

Figura 7 pode-se observar essa organização hierárquica das necessidades no formato de uma árvore de objectivos.

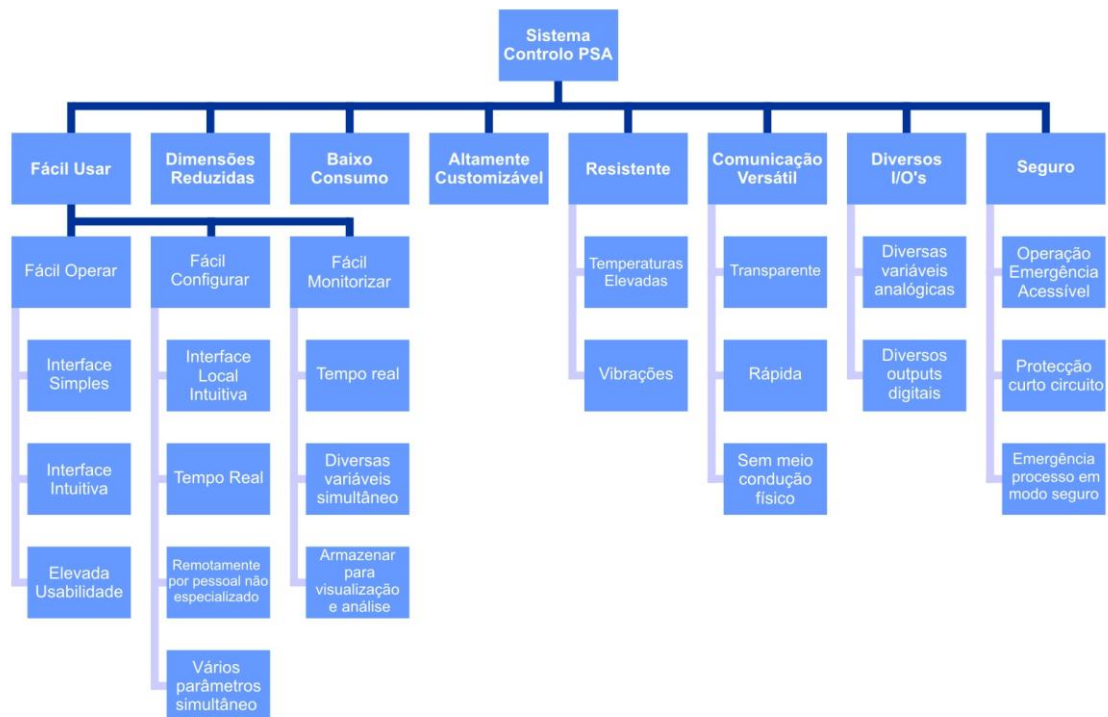


Figura 7 - Diagrama hierárquico de necessidades do cliente

3.1.3. Determinação da importância

Foram atribuídas prioridades às diversas necessidades, sendo de destacar as seguintes:

- O sistema deverá ser fácil de usar, tanto na vertente de configuração, como na de operação e monitorização, independentemente do tipo de utilizador final;
- O sistema deverá ser altamente parametrizável local e remotamente;
- A comunicação com o pc remoto deverá ser feita da forma mais versátil possível, através de uma ligação sem fios de modo a ser transparente para os utilizadores.

3.2. Requisitos Técnicos

Um dos *inputs* para a especificação de requisitos é o ambiente, as suas restrições e *standards*. Uma restrição é um tipo especial de requisito, uma decisão de projecto imposta pelo ambiente que limita o projecto e especifica como o sistema será implementado.

Algumas decisões previamente tomadas pelo cliente, baseadas na sua experiência técnica e conhecimento de mercado, são apresentadas de seguida:

- O bloco central responsável pelo controlo de todo o sistema deverá ser um autómato da marca OMRON, modelo CJ1M-CPU11;
- Esse autómato modular tem de usar as cartas de expansão CJ1W-ID211 para entradas digitais, CJ1W-OC211 para saídas digitais e CJ1W-AD081-V1 para entradas analógicas;
- A interface com o operador deverá ser implementada com uma consola gráfica da OMRON, modelo NS5-SQ00B-V2;
- As comunicações sem fios deverão ser implementadas com recurso ao modem GPRS Westermo GDW-11. Após uma revisão dos requisitos este requisito técnico foi substituído por um requisito funcional: as comunicações deverão ser implementadas por um modem GPRS que funcione tanto no continente europeu como no americano.

Capítulo 4

Solução Tecnológica

4.1. Arquitectura do sistema

Com base nos requisitos e necessidades agrupados escolheu-se a arquitectura para o sistema apresentada na tabela 1.

Tabela 1 - Arquitectura de alto nível do sistema

Controlo central	Comunicações	Interface Local	Interface Remota	Monitorização do Processo
Autómato Programável (PLC)	GSM/GPRS	Consola gráfica HMI + Botões	PC remoto com aplicação visual de parametrização e monitorização	Dispositivos de Medida e Monitorização (DMM)

4.2. Design de sistema

4.2.1. Decomposição Funcional

Apresenta-se de seguida o design do sistema de controlo da unidade de PSA. Foi usada uma abordagem *top-down* para a sintetização do design do sistema, começando por um

primeiro nível mais abstracto, apresentado na Figura 8, que representa a visão geral do sistema, as suas entradas e saídas e os seus requisitos funcionais, isto é o que o sistema terá de fazer.

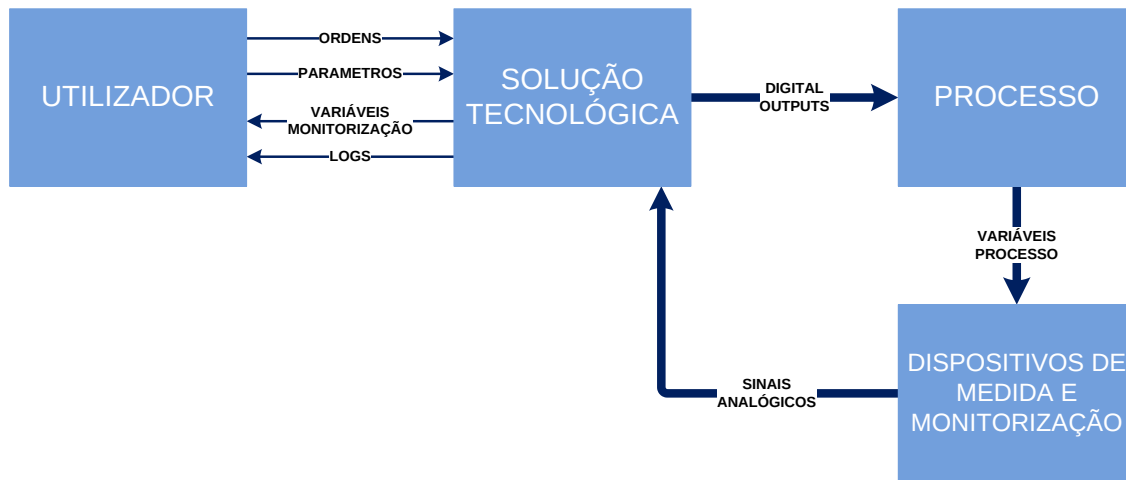


Figura 8 - Diagrama funcional de alto nível do sistema de controlo - vista do hardware

A solução tecnológica encontra-se representada como um bloco funcional fechado com entradas e saídas que proporcionam interacção com outros blocos funcionais exteriores ao sistema de controlo:

- Processo - representa a unidade de PSA e inclui toda a dinâmica do processo de PSA e o todo o equipamento físico do qual se deve destacar as válvulas, o compressor, a bomba de vácuo e a ventilação;
- Dispositivos de medida e monitorização (DMM) - representa os vários dispositivos aplicados no processo responsáveis por medirem e monitorizarem as variáveis de processo e enviarem as mesmas para o sistema de controlo;
- Utilizador - bloco de alto nível que representa os utilizadores que possam interagir com o sistema de controlo, a nível operacional, monitorização ou parametrização.

Uma nova iteração permitiu subdividir os blocos anteriores e levou à criação de um segundo nível mais pormenorizado apresentado na

Figura 9, com a representação dos componentes do sistema, agrupados por funcionalidade, as interacções entre eles e os seus requisitos funcionais.

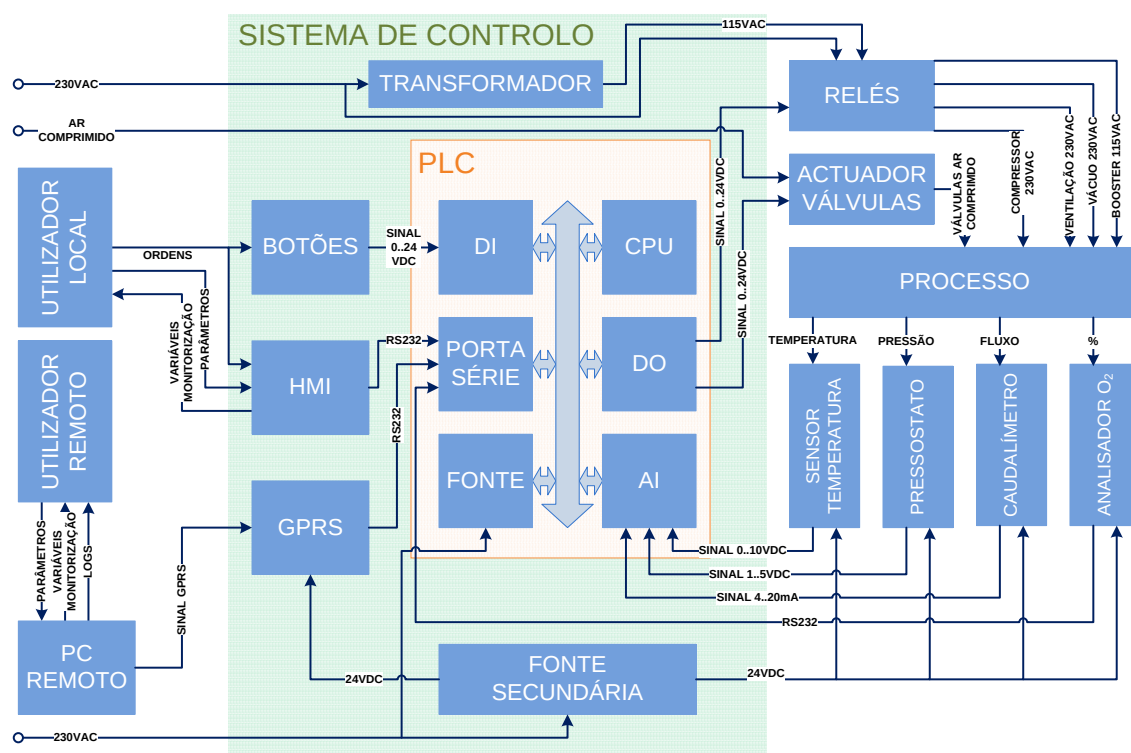


Figura 9 - Diagrama funcional de baixo nível do sistema de controlo - vista do hardware

O bloco do sistema de controlo encontra-se dividido em vários blocos funcionais mais específicos detalhando dessa forma a arquitectura do hardware do sistema de controlo o que permite uma implementação mais eficaz. Segue-se uma breve lista com clarificação dos blocos mais importantes:

- Macro bloco PLC - bloco principal do sistema de controlo. Subdivide-se em 6 blocos funcionais que comunicam entre si por um barramento de dados. O bloco CPU é responsável pela execução de instruções, cálculos e pelo controlo de todos os outros blocos. A fonte de tensão é responsável pela alimentação eléctrica do PLC. Os blocos DI, AI e DO tratam respectivamente das entradas digitais (0..24VDC), entradas analógicas (0..10VDC, 1..5VDC e 4..20mA) e saídas digitais (0..24VDC).
- Válvulas pneumáticas - recebem sinais de comando (0..24VDC) do PLC e usando o ar comprimido abrem ou fecham as válvulas de acordo com esses sinais.
- Relés - cortam ou fornecem a alimentação ao compressor, bomba de vácuo, ventilação e *booster*.
- DMM - subdivididos em 4 blocos. São eles os transdutores de temperatura, sensores de pressão, caudalímetros e analisadores de oxigénio.
- Utilizadores - identificam-se dois blocos distintos, o utilizador que usa o PC remoto para comunicar com o PLC através de GPRS e o utilizador local usa uma interface HMI e um conjunto de botões com as operações essenciais de controlo do processo.

Uma parte extremamente importante do sistema é o software de controlo e a respectiva parametrização.

Apresenta-se na Figura 10 o diagrama detalhado do software, evidenciando os seus módulos funcionais.

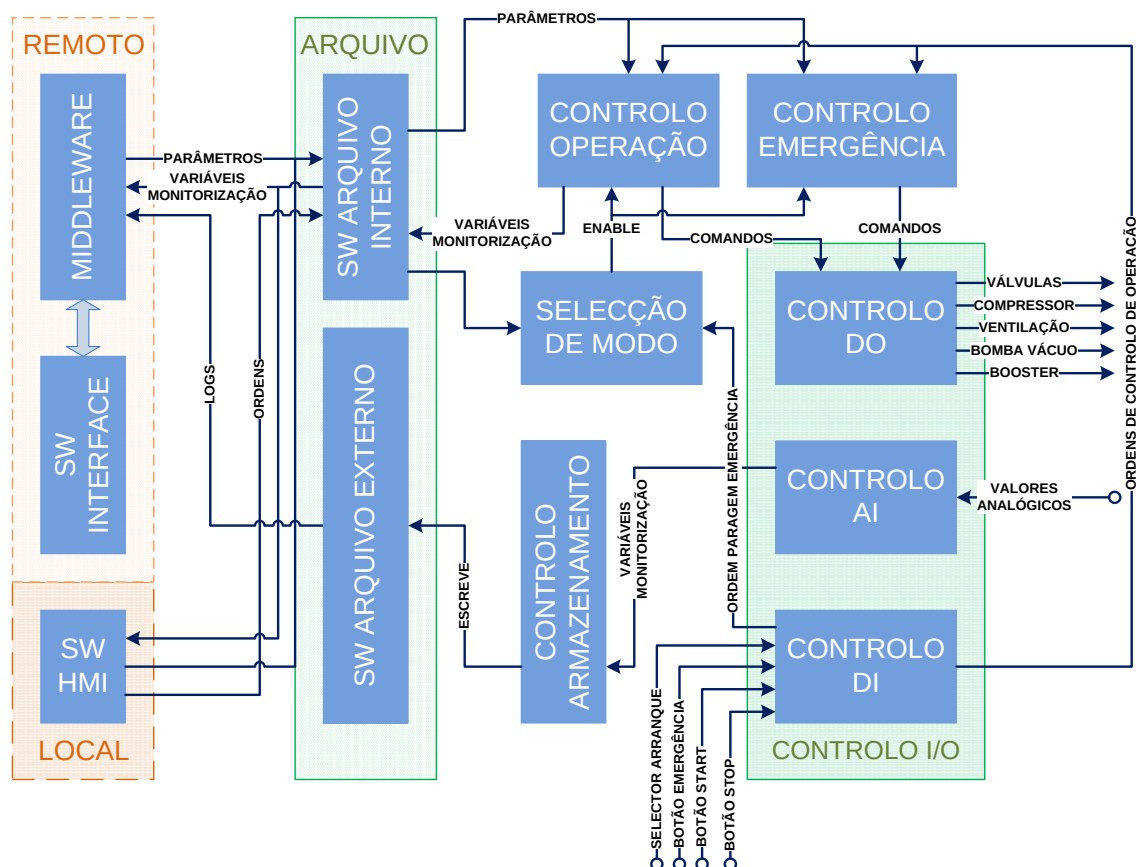


Figura 10 - Diagrama funcional de baixo nível do software de controlo

Identificam-se imediatamente três módulos distintos de software. O software de controlo que é executado no PLC, o software que é executado na interface homem máquina e por fim o conjunto de software executado no PC remoto que comunica com o PLC através de um modem GPRS.

No PC remoto foi criada uma aplicação visual e intuitiva que tem como função parametrizar o sistema, mais especificamente o nº de etapas em cada macro-etapa, o tempo de duração de cada etapa, o nº de ciclos de cada macro-etapa e quais as saídas digitais a activar em cada etapa. Deverá também ser usada para visualizar em tempo real as variáveis monitorizadas e observar o comportamento temporal dessas variáveis guardando um registo em memória externa do PLC. De forma a simplificar essa aplicação, é usado um “middleware”

que implementa o protocolo de comunicações com o PLC e permite aceder às suas posições de memória de forma simples e transparente.

O software de controlo divide-se em cinco blocos funcionais:

- Selecção de modo - activa o módulo de controlo de emergência se alguma ordem de paragem de emergência for identificada enquanto o processo de PSA está activo;
- Controlo de operação - controla as transições entre as etapas dos ciclos de produção, o número de ciclos dentro de cada macro-etapa, os tempos de activação de cada etapa de acordo com um conjunto de parâmetros armazenados na memória do PLC e envia comandos para o controlo de I/Os;
- Controlo de emergência - usado para colocar o sistema num estado seguro, controla as etapas de um ciclo de término de acordo com um conjunto de parâmetros armazenados na memória do PLC e, tal como o módulo anterior, envia comandos para o controlo de I/Os;
- Controlo de armazenamento - amostra periodicamente um conjunto de variáveis de monitorização e armazena o seu valor na memória externa do PLC.
- Controlo de I/Os - este último módulo subdivide-se noutros três módulos:
 - Controlo DI - analisa as entradas digitais do PLC e de acordo com estas envia ordens para os outros blocos, por exemplo a ordem de paragem de emergência para o bloco de selecção de modo;
 - Controlo AI - amostra as entradas analógicas periodicamente, dimensiona o valor das entradas de acordo com dados de calibração existentes em memória e armazena os valores finais em posições de memória pré-definidas para posterior monitorização e controlo;
 - Controlo DO - usa os comandos enviados pelo controlo de operação e pelo controlo de emergência para identificar qual a posição de memória onde estão armazenados os valores a colocar nas saídas digitais.

4.2.2. Modelos de comportamento

Com o objectivo de estruturar e melhorar a fase de criação do software de controlo do processo de PSA, foi desenvolvido um modelo do software em UML. Os diagramas UML oferecem uma forma *standard* de visualizar a arquitectura do sistema. De seguida serão apresentados os diagramas UML criados para descrever os blocos funcionais de software controlo de operação, selecção de modo e controlo de emergência (sob a forma de um ciclo de controlo genérico).

O bloco mais geral, responsável por controlar o modo em que o sistema se encontra, é o bloco de *selecção de modo*, apresentado na Figura 11.

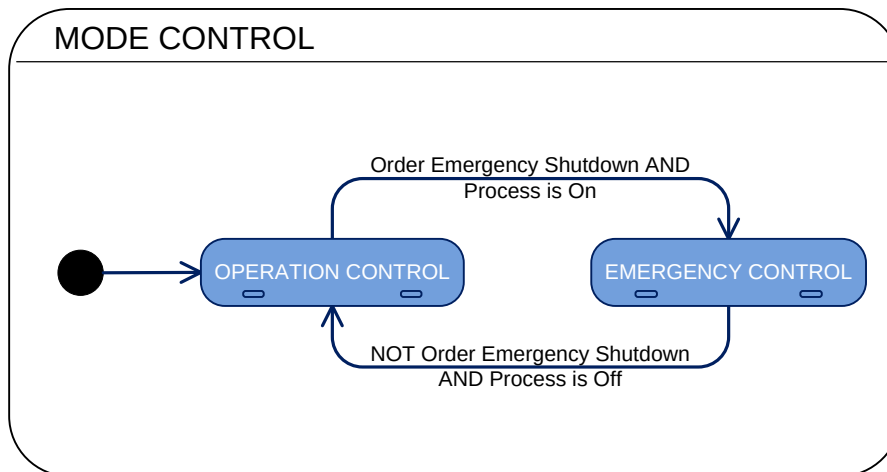


Figura 11 - Diagrama de estados UML do bloco selecção de modo

Quando o PLC é ligado passa automaticamente para o modo de operação, e as saídas do sistema são controladas pelo bloco *controlo de operação*. No caso do botão de emergência ser premido enquanto o sistema estiver a produzir, o controlo de operação é desligado e o sistema entra em controlo de emergência. Apenas volta ao estado normal quando o botão de emergência for desligado e acabar o ciclo de término de emergência.

Um sistema com memória é capaz de mudar a resposta a um mesmo conjunto de entradas baseando-se no estado anterior do sistema. Quando o processo a modelizar é de um sistema com memória podem usar-se diagramas de estado UML para descrever o seu comportamento.

É o caso específico do controlo de processo de PSA, este sistema é orientado a eventos com memória, Figura 12.

Quando o bloco controlo de operação é activado e dependendo do modo de arranque seleccionado o sistema será colocado em *standby* seguro ou normal. É possível dar ordem de manutenção ou ordem de arranque ao sistema. Se for dada ordem de arranque, dependendo do estado em que se encontra o sistema, o estado seguinte será o macro-estado de inicialização segura ou normal. Estes macro-estados são encapsulamentos de conjuntos de outros estados. O algoritmo executado no interior desses macro-estados será discutido posteriormente. Os macro-estados são executados sequencialmente até se atingir o estado de produção. O sistema ficará continuamente neste estado até ser dada ordem de fim de operação, após a qual serão executados dois ciclos de término e o sistema colocado em *standby* normal.

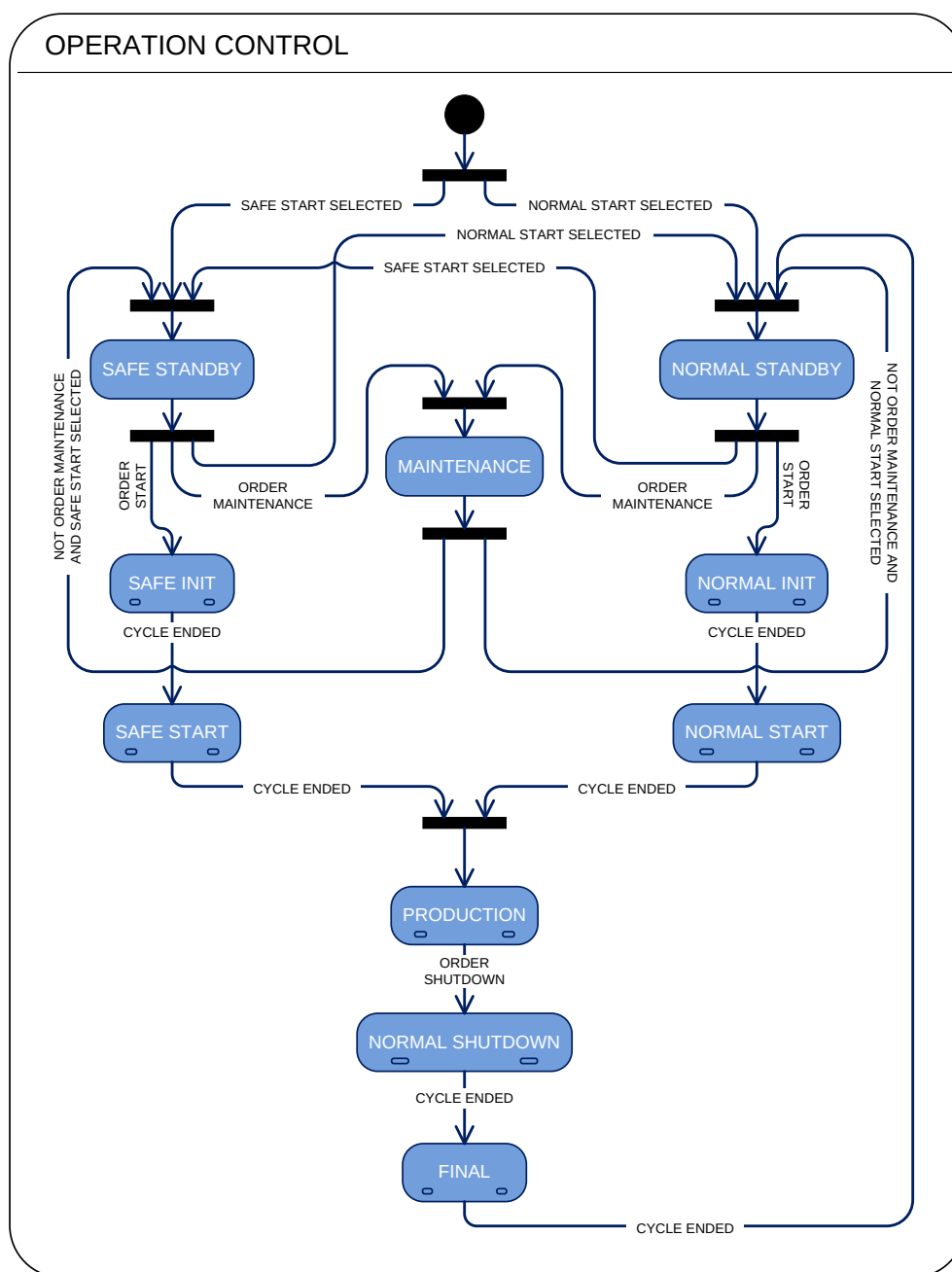


Figura 12 - Diagrama de estados UML do bloco controlo de operação

Tal como mostrado em capítulos anteriores, um processo PSA simplificado é constituído por duas fases, adsorção e regeneração. No entanto, outras abordagens de design são possíveis e usualmente têm mais etapas que o processo simplificado. A abordagem usada neste projecto considera a adição de mais duas pré-colunas, às duas colunas já existentes no processo simplificado, para remoção de água e dióxido de carbono e uma sequência de ciclo com as seguintes etapas:

- Pressurização parcial com igualização *Top-Bottom*;
- Pressurização parcial com produto (*backfill*);
- Pressurização com ar comprimido;
- Produção.

- Despressurização parcial com igualização *Top-Bottom*;
- *Blow-down*
- Regeneração em vácuo com fluxo de produto em contra-corrente.

Uma vez que o próprio processo a ser controlado por este sistema também é um projecto experimental, assume-se que o número de estados poderá variar dentro de um intervalo conhecido (1 a 12 estados). É por isso de capital importância que o software tenha um número livre de estados cujo valor seja um parâmetro configurável.

Todos os macro-estados executam então um mesmo ciclo de controlo genérico com 12 estados cujo modelo se encontra na Figura 13. A diferença entre cada um deles está declarada em variáveis armazenadas em posições de memória do PLC. Deste modo são configurados os tempos e válvulas que se pretendem usar deixando os restantes com tempo de duração igual a zero segundos.

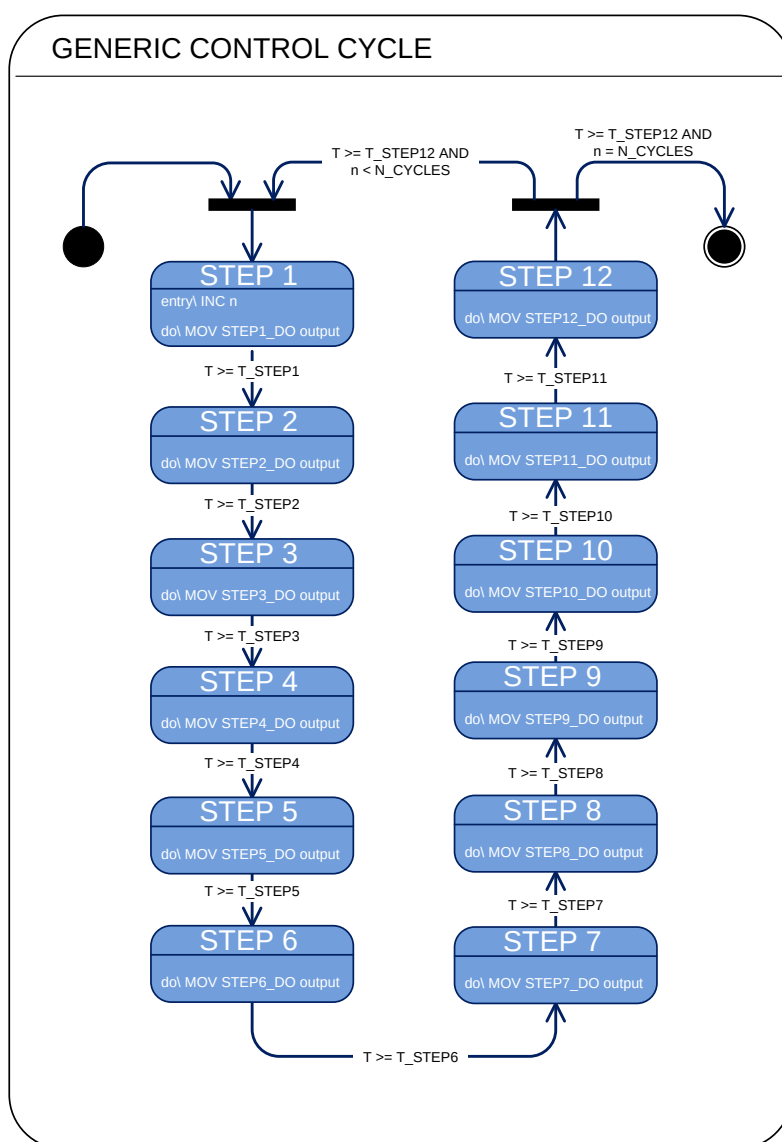


Figura 13 - Diagrama de estados UML de um bloco de controlo de ciclo com 12 etapas genérico

Capítulo 5

Protótipo

5.1. Hardware

Tal como visto no capítulo anterior, o hardware usado neste sistema é composto por um PLC, um módulo de comunicações GPRS e uma consola HMI que serão descritos de seguida.

5.1.1. PLC

Um dos requisitos técnicos do projecto era o uso de um PLC da OMRON, mais especificamente o CJ1M CPU11, mostrado na Figura 14. Já durante a fase de prototipagem esse CPU foi trocado por outro da mesma família mas com mais memória, o CJ1M CPU13. As razões que levaram a essa troca são apresentadas na secção 5.2 relativa a software.

Este PLC apresenta uma forma modular, diminuindo custos com futuras actualizações, aumentando o grau de expansão e personalização de sistemas, mantendo o design do sistema de controlo.



Figura 14 - PLC Omron CJ1M CPU11

Das suas características técnicas destacam-se:

- PLC modular ao nível de entrada de gama;
- CPUs com I/O's de impulsos incorporadas para controlo do movimento ou com interface de Ethernet para fácil integração;
- Programação em linguagem de texto estruturado com base na norma IEC 61131-3, extensas bibliotecas de blocos de funções;
- Encaminhamento de comunicações transparente através de redes diferentes;
- Ranhura da placa de memória CompactFlash para armazenamento de dados e troca de programas.

Foram usados os seguintes módulos de entradas e saídas digitais e analógicas:

Tabela 2 - Entradas e saídas digitais do PLC Omron

ID	Pontos	Tipo	Tensão Nominal	Corrente Nominal
CJ1W-ID211	16	Entrada DC	24 VDC	7 mA
CJ1W-OC211	16	Saída relé	250 VAC	2 A

Tabela 3 - Entradas e saídas analógicas do PLC Omron

ID	Pontos	Tipo	Gama	Resolução	Tempo de Conversão
CJ1W-AD081-V1	8	Entrada analógica	1 a 5V 0 a 10V -10 a 10V 1 a 5V 4 a 20mA	1/8000	250 µs/ponto

5.1.2. Comunicação GPRS

Para implementar as comunicações GPRS no sistema de controlo, foi escolhido o modem OnCell G2110T, Figura 15, que pertence a uma série de modems quad-band GSM/GPRS para uso industrial, capazes de transmitir dados e mensagens de texto (SMS) em redes móveis GSM/GPRS. Das características principais, salienta-se a montagem em calha DIN, o funcionamento com tensões de 12 a 48 VDC e a protecção contra sobrecargas nas portas série. Adicionalmente, o modem OnCell G2110-T permite suportar uma gama de temperaturas mais elevada (-40 to 75 °C) tornando-o adequado a ambientes industriais.



Figura 15 - Modem Moxa OnCell G2110T

5.1.3. Human-Machine Interface (HMI)

Para interface entre o sistema de controlo e o operador foi seleccionada a consola HMI Omron NS5, apresentada na Figura 16.

Este modelo possibilita um máximo de 256 cores e está equipado com uma ligação USB para transferir/enviar projectos e a possibilidade de comunicar através de Ethernet. Outras vantagens enunciadas na folha do produto são:

- Clareza perfeita e ecrãs de comutação rápida;
- Duração da luz de fundo extremamente longa (até 75.000 horas);
- Suporta todos os idiomas da Europa, Ásia e Cirílico;
- Fácil registo de dados numa memória CompactFlash;
- Tamanho de memória grande (20 MB).



Figura 16 - Consola HMI OMRON NS5

5.1.4. Sistema de controlo

Para além do hardware apresentado nos pontos anteriores, foram adquiridos outros equipamentos, após exaustiva procura e comparação de várias soluções no mercado, para implementação dos restantes blocos funcionais do sistema de controlo. O hardware foi montado no interior de um armário metálico *standard*, Figura 17, e aplicado em calhas DIN para fácil reconfiguração.

Todo o equipamento responsável por entradas ou saídas do sistema foi montado na parte inferior do armário. Uma vez que se trata de um protótipo, que se sofrerá alterações constantes, os cabos de alimentação, sinal e ar comprimido foram instalados com algum excesso de comprimento de forma a reduzir a carga de trabalho necessária para uma reconfiguração de ligações resultante de uma instalação num armário de medidas diferentes.

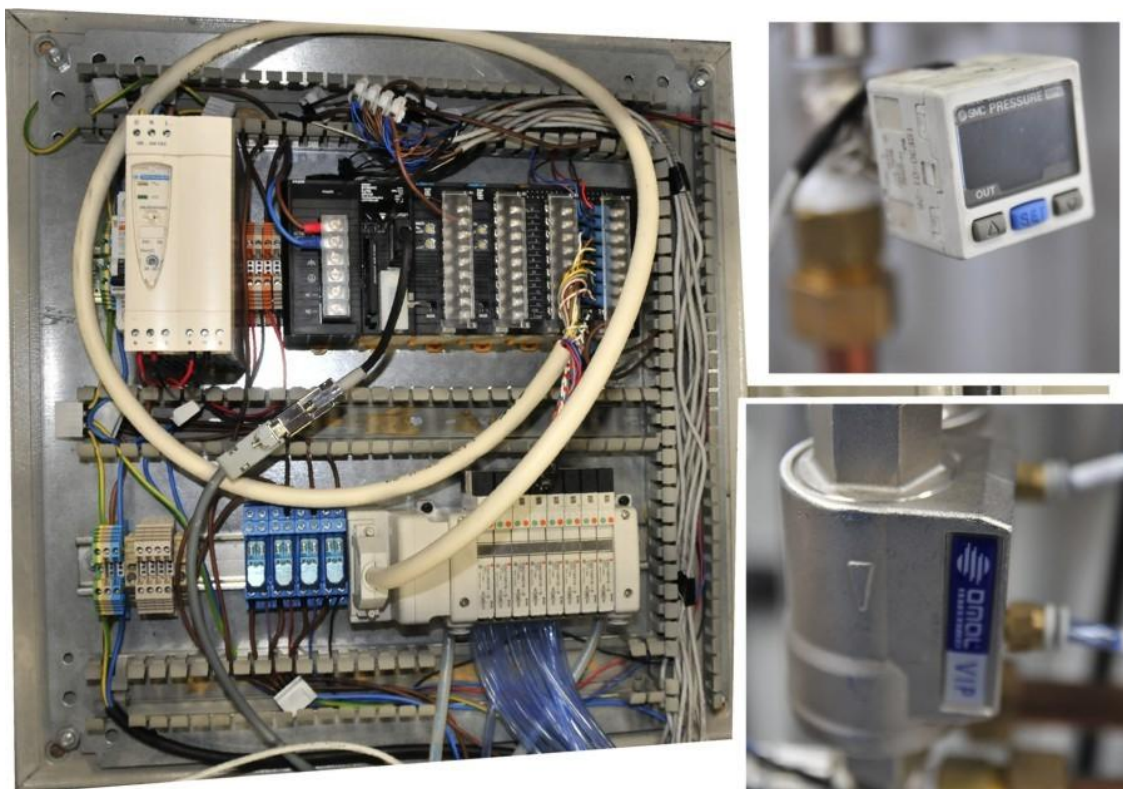


Figura 17 - Montagem do sistema de controlo no armário eléctrico (esquerda); pressostato digital e válvula (direita)

O sistema foi protegido com um disjuntor instalado imediatamente ao lado da fonte de alimentação secundária e possui também um botão exterior de fácil acesso para ligar ou desligar o sistema. A consola HMI foi instalada na porta para servir de interface com o

operador e adicionados botões para as funções principais de controlo do sistema na parte inferior da porta. O aspecto final do armário pode ser visto na Figura 18.



Figura 18 - Vista do armário do sistema de controlo com porta fechada

Na Figura 19 pode-se observar o protótipo durante um dos testes de controlo do processo de PSA.

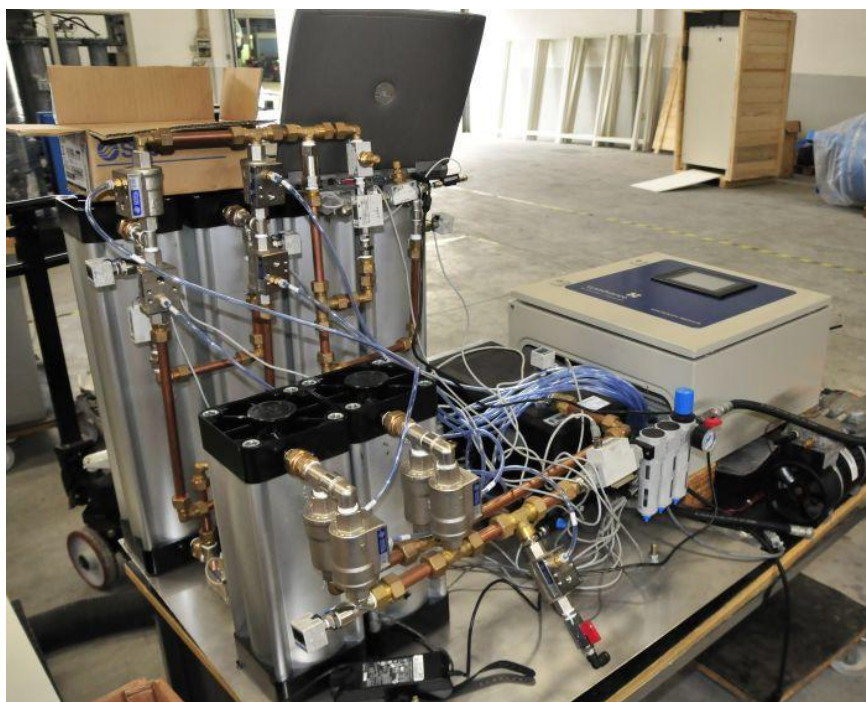


Figura 19 - Unidade PSA experimental e sistema de controlo

5.2. Software

5.2.1. Software usado

OMRON CX-Programmer

Ferramenta de programação para o autómato programável (PLC) utilizado em todos os modelos da Omron. Inclui caixas de diálogo na definição de parâmetros para minimizar o tempo de configuração. Blocos de funções standard em texto estruturado em conformidade com CEI 61131-3 ou numa linguagem em "ladder" convencional que simplificam o desenvolvimento de programas. A ligação ao PC para programação pode ser feita através de USB ou ligações série.

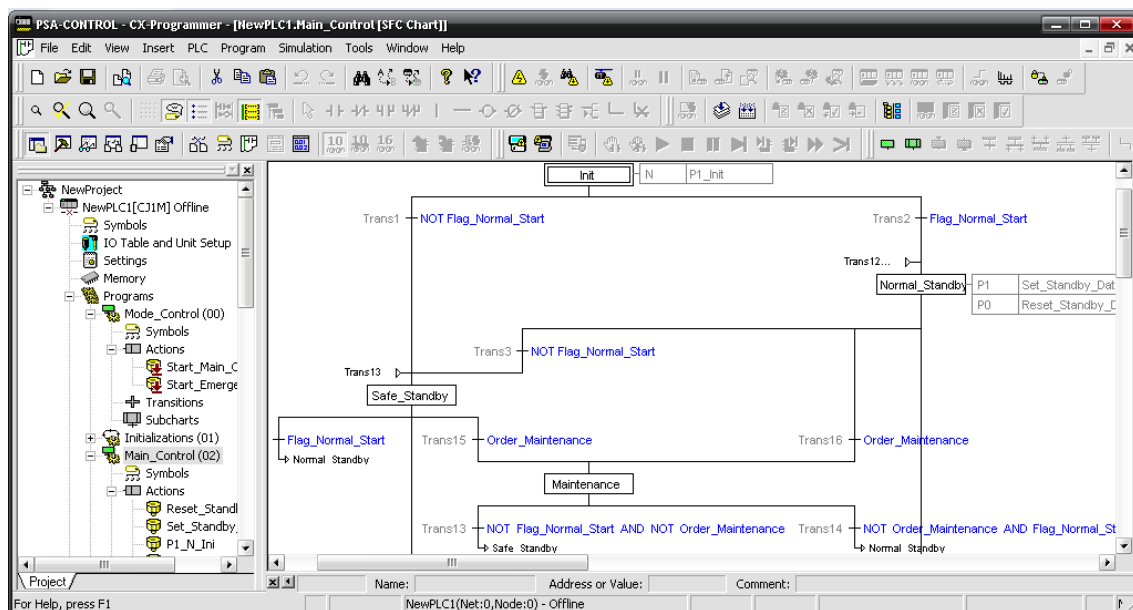


Figura 20 - Software CX Programmer da Omron

OMRON CX-Designer

Software de desenho utilizado para programar consolas HMI da Omron; neste projecto específico a consola NS5. Apresenta uma interface com o utilizador totalmente personalizável com ícones para a maior parte das funções, permitindo a reutilização de projectos e ecrãs pela funcionalidade *drag & drop*, ou por exportação/importação de variáveis.

Podem-se partilhar variáveis entre o PLC e a HMI, evitando duplicação das mesmas; basta arrastar e largar a partir do CX-Programmer ou copiar e colar a partir do Excel.

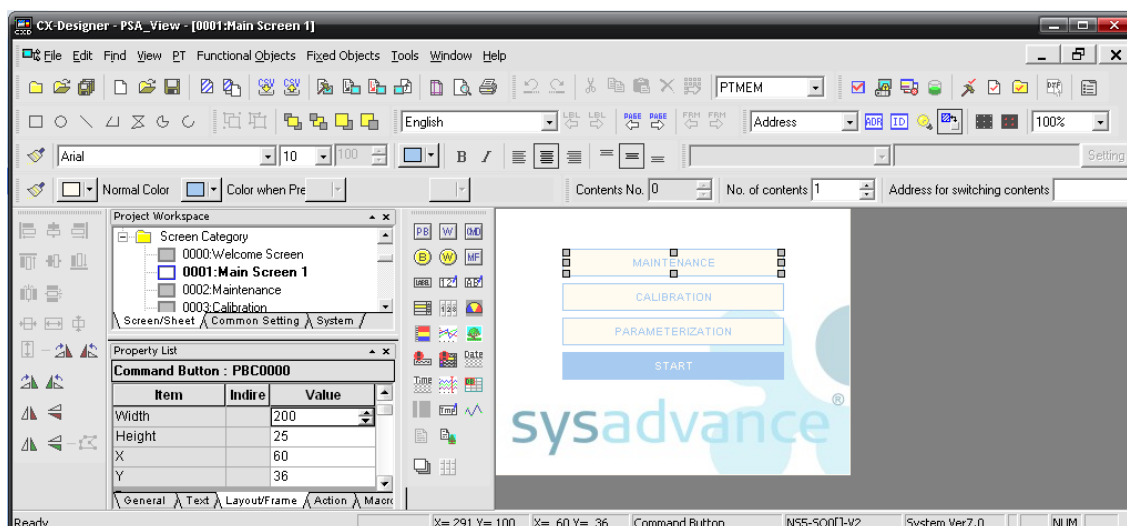


Figura 21 - Software CX Designer da Omron

OMRON CX-Server Lite

Este *middleware* foi utilizado para criar projectos de ligação entre o PLC e o PC remoto, permitindo a recepção e o envio de dados de e para o PLC. Permite a programação usando o Visual Basic, Excel ou .NET, oferecendo uma configuração simples através de componentes gráficos ou linguagem de extensão complexa com API.

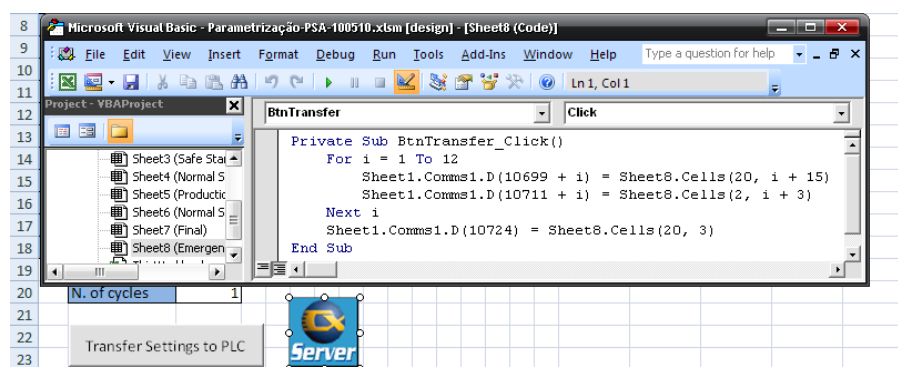


Figura 22 - Componentes CX-Server Lite da Omron em ambiente Excel+VBA

5.2.2. Programação

Para a programação do software do CJ1M usou-se, tal como indicado na subsecção anterior, a aplicação *CX-Programmer* da Omron.

Para cada bloco funcional foi escolhida a linguagem de programação do IEC 61131 que melhor se adaptasse ao comportamento do processo que esse bloco controla.

Sequential Function Chart ou SFC foi usado para a estruturação e programação dos blocos controlo de operação, controlo de emergência e selecção de modo. A mesma justificação do uso de diagramas de estado UML para a modelização do controlo do processo é aplicada para a escolha de SFC como linguagem de programação destes módulos, isto é, trata-se de um

sistema com memória e de um processo sequencial. A sua representação gráfica de estados e transições permite facilmente executar a passagem de modelos UML de software e/ou fluxogramas de processo para código a executar no PLC.

A correspondência quase exacta entre o código em SFC e o modelo do processo, como se pode observar na Figura 23, torna intuitiva a compreensão do código mesmo por programadores que não tenham desenvolvido o código facilitando assim o debug e a manutenção.

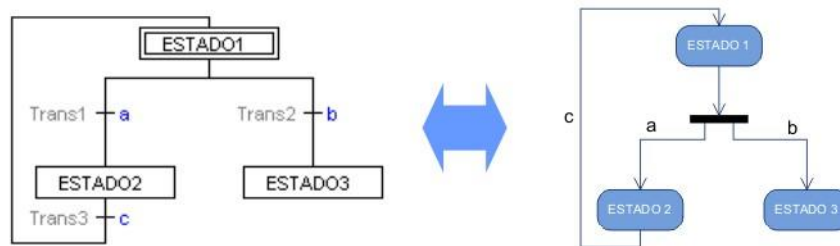


Figura 23 - Comparação entre linguagem de programação SFC e diagrama de estados UML

O resultado da programação do controlo de operação e outros módulos em SFC é visualmente semelhante ao seu modelo UML como se pode verificar na Figura 24.

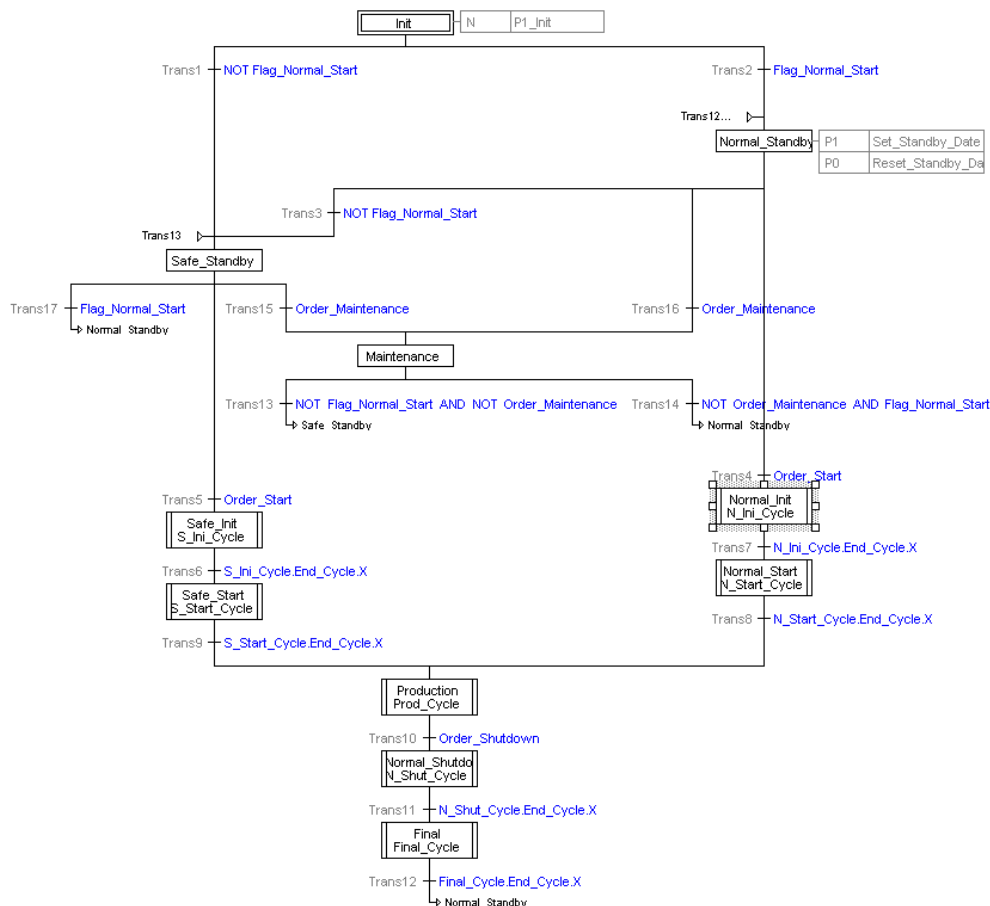


Figura 24 - Vista do código de controlo principal em SFC

Ladder ou LD foi usado para os blocos controlo de I/Os, controlo de armazenamento e para diversas acções executadas pelos estados da linguagem SFC. O LD foi escolhido para esta tarefa porque é uma linguagem simples e adaptada para controlo de inputs e outputs binários, mas principalmente porque é uma linguagem muito suportada pela OMRON, com extensa documentação e exemplos.

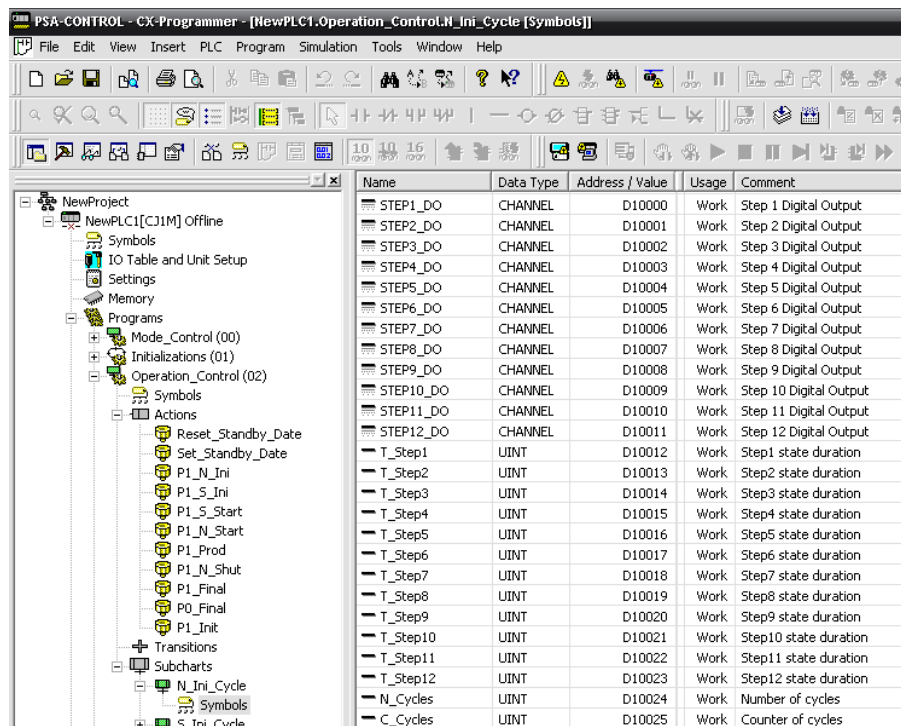
O bloco de controlo de DO foi implementado em *Ladder* para reduzir a quantidade de memória de programa do PLC consumida. O autómato usado no início do projecto, OMRON CJ1M-CPU11, possui uma memória de programa de 5 Ksteps. À medida que o projecto evoluiu e novos requisitos foram acrescentados, o número de estados aumentou e a memória disponível reduziu-se até se esgotar.

Um pequeno cálculo explica rapidamente como se esgotou a capacidade do PLC. O programa principal de controlo tem 8 macro-estados, com 12 sub-estados cada um, o que resulta num total de 96 estados e o mesmo número de acções associadas a cada estado. Apenas a associação de uma acção em *Ladder* vazia (sem código no interior da acção) a um estado ocupa cerca de 50 Steps. Associando as 96 acções ficamos com 4800 Steps de memória ocupados, quase a totalidade da memória do CPU.

A solução passou por fazer alterações ao hardware e ao software. O CPU foi trocado para o CJ1M-CPU13 com 20 Ksteps de memória de programa. A estrutura do programa continuou em SFC, mas associaram-se acções booleanas, que ocupam apenas 2 Steps cada, aos estados. O controlo de DO foi implementado em LD recorrendo a endereçamento indirecto, resultando numa poupança significativa da memória.

Structured Text (ST) foi usado para dois blocos, não representados nos diagramas de decomposição funcional, que executam cálculos aritméticos para outros blocos funcionais, mais especificamente o cálculo do *offset* a ser usado pelo endereçamento indirecto do bloco controlo de DO e o cálculo do modo de arranque do controlo de operação. A facilidade em implementar operações matemáticas complexas com esta linguagem foi a razão para a sua escolha.

Durante a fase de programação foi realizado um esforço para comentar todo o código gerado e atribuir nomes significativos a cada variável e estado de forma a facilitar o processo de manutenção. Foi evitado também o uso de valores absolutos em todo o código, usando como alternativa variáveis para definir tempos de duração de estados, curvas de calibração de sensores, número de etapas e ciclos por cada macro-etapa, de forma a obter um programa final completamente parametrizável. Uma listagem exemplo encontra-se na Figura 25.



The screenshot shows the 'PSA-CONTROL - CX-Programmer' interface. The left pane displays a project tree with the following structure:

- NewProject
 - NewPLC1[CJ1M] Offline
 - Symbols
 - IO Table and Unit Setup
 - Settings
 - Memory
 - Programs
 - Mode_Control (00)
 - Initializations (01)
 - Operation_Control (02)
 - Symbols
 - Actions
 - Reset_Standby_Date
 - Set_Standby_Date
 - P1_N_Ini
 - P1_S_Ini
 - P1_S_Start
 - P1_N_Start
 - P1_Prod
 - P1_N_Shut
 - P1_Final
 - P0_Final
 - P1_Init
 - Transitions
 - Subcharts
 - N_Ini_Cycle
 - Ini_Cycle

The right pane displays a table of variables used in the parametrization of one of the macro-states:

Name	Data Type	Address / Value	Usage	Comment
STEP1_DO	CHANNEL	D10000	Work	Step 1 Digital Output
STEP2_DO	CHANNEL	D10001	Work	Step 2 Digital Output
STEP3_DO	CHANNEL	D10002	Work	Step 3 Digital Output
STEP4_DO	CHANNEL	D10003	Work	Step 4 Digital Output
STEP5_DO	CHANNEL	D10004	Work	Step 5 Digital Output
STEP6_DO	CHANNEL	D10005	Work	Step 6 Digital Output
STEP7_DO	CHANNEL	D10006	Work	Step 7 Digital Output
STEP8_DO	CHANNEL	D10007	Work	Step 8 Digital Output
STEP9_DO	CHANNEL	D10008	Work	Step 9 Digital Output
STEP10_DO	CHANNEL	D10009	Work	Step 10 Digital Output
STEP11_DO	CHANNEL	D10010	Work	Step 11 Digital Output
STEP12_DO	CHANNEL	D10011	Work	Step 12 Digital Output
T_Step1	UINT	D10012	Work	Step1 state duration
T_Step2	UINT	D10013	Work	Step2 state duration
T_Step3	UINT	D10014	Work	Step3 state duration
T_Step4	UINT	D10015	Work	Step4 state duration
T_Step5	UINT	D10016	Work	Step5 state duration
T_Step6	UINT	D10017	Work	Step6 state duration
T_Step7	UINT	D10018	Work	Step7 state duration
T_Step8	UINT	D10019	Work	Step8 state duration
T_Step9	UINT	D10020	Work	Step9 state duration
T_Step10	UINT	D10021	Work	Step10 state duration
T_Step11	UINT	D10022	Work	Step11 state duration
T_Step12	UINT	D10023	Work	Step12 state duration
N_Cycles	UINT	D10024	Work	Number of cycles
C_Cycles	UINT	D10025	Work	Counter of cycles

Figura 25 - Lista de variáveis usadas na parametrização de um dos macro-estados

PC REMOTO

Foi desenvolvida uma aplicação em Microsoft Excel + VBA para a parametrização do sistema. A escolha do Microsoft Excel preenche um dos requisitos do projecto, obter uma interface intuitiva e que o operador remoto tenha facilidade de utilizar, independentemente de ter conhecimentos ou não na área de programação de autómatos.

A aplicação permite, para cada macro-estado do processo, que o utilizador defina durante quanto tempo é que cada sub-estado estará activo, o estado a cada instante das válvulas, compressor, booster, bomba de vácuo, ventilação e por fim o número de vezes que cada macro-estado é executado até passar ao seguinte. O aspecto da aplicação desenvolvida para interface remota está apresentado na Figura 26.

	A	B	C	D	F	H	J	L	N	P	R	T	V	X	Z	AB
1			Step	1	2	3	4	5	6	7	8	9	10	11	12	
2	Ch.	Name	time (s):	2	2	10	14	2	2	10	14	0	0	0	0	
3	0	Ventilation		1	1	1	1	1	1	1	1	0	0	0	0	
4	1	Vaccum Pump		1	1	1	1	1	1	1	1	0	0	0	0	
5	2	Compressor		1	1	1	1	1	1	1	1	0	0	0	0	
6	3	Booster		1	1	1	1	1	1	1	1	0	0	0	0	
7	4	Valve K01 N/C		1	1	1	1	0	0	0	0	0	0	0	0	
8	5	Valve K02 N/C		0	0	0	0	1	1	1	1	0	0	0	0	
9	6	Valve K03 N/C		1	0	0	0	1	1	1	1	0	0	0	0	
10	7	Valve K04 N/C		1	1	1	1	1	0	0	0	0	0	0	0	
11	8	Valve K05 N/C		0	0	1	1	0	1	1	1	0	0	0	0	
12	9	Valve K06 N/C		0	1	1	1	0	0	1	1	0	0	0	0	
13	10	Valve K07 N/C		0	0	0	0	1	0	0	0	0	0	0	0	
14	11	Valve K08 N/C		1	0	0	0	0	0	0	0	0	0	0	0	
15	12	Valve K09 N/C		0	1	1	1	0	0	0	0	0	0	0	0	
16	13	Valve K10 N/C		0	0	0	0	0	1	1	1	0	0	0	0	
17	14	Valve K11 N/C		1	1	1	1	0	1	1	1	0	0	0	0	
18	15	Valve K12 N/O		1	1	1	1	1	1	1	1	0	0	0	0	
19																
20		N. of cycles	-1													
21																
22																
23																
24																
25																
26																
27																
28																
29																
30																
31																
32																
33																
34																
35																
36																
37																
38																
39																
40																
41																
42																
43																
44																
45																
46																
47																
48																
49																
50																
51																
52																
53																
54																
55																
56																
57																
58																
59																
60																
61																
62																
63																
64																
65																
66																
67																
68																
69																
70																
71																
72																
73																
74																
75																
76																
77																
78																
79																
80																
81																
82																
83																
84																
85																
86																
87																
88																
89																
90																
91																
92																
93																
94																
95																
96																
97																
98																
99																
100																

Figura 26 - Aplicação de parametrização remota em Microsoft Excel

É usado o middleware CX-Server Lite que disponibiliza um conjunto de objectos para VBA que estabelecem uma ligação entre esta aplicação em Excel e o PLC remoto.

HMI

A última aplicação a ser desenvolvida no âmbito do projecto foi o software de monitorização que será executado na consola HMI.

São apresentados de seguida alguns dos sinópticos criados e a sua utilidade:

- Ecrã de apresentação - um simples ecrã de boas vindas, desaparece quando tocado;



Figura 27- Sinóptico de apresentação

- Menu principal - tem dois estados possíveis, quando o processo está em *standby* e quando está *online*. Em *standby* o operador tem a escolha de iniciar o processo de produção pressionando *START* ou entrar nos menus de manutenção, calibração e parametrização. No modo *online* terá mais algumas escolhas disponíveis. Pode desligar o processo em *SHUTDOWN* ou entrar nos menus de alarmes, calibração e parametrização em tempo real, gráficos e sinóptico de visualização do processo de PSA.



Figura 28 - Sinóptico do menu principal

- Manutenção - este ecrã permite ao operador abrir ou fechar manualmente cada uma das válvulas do sistema para despistar potenciais avarias;



Figura 29 - Sinóptico de manutenção

- Parametrização - permite a configuração dos tempos de duração de cada sub-estado do processo de PSA;

INPUT NEW STEP TIME VALUES			
STEP1	-99999	STEP7	-99999
STEP2	-99999	STEP8	-99999
STEP3	-99999	STEP9	-99999
STEP4	-99999	STEP10	-99999
STEP5	-99999	STEP11	-99999
STEP6	-99999	STEP12	-99999
UPDATE			

Figura 30 - Sinóptico de parametrização

- Gráficos - o operador tem a possibilidade de visualizar o valor de cada entrada analógica do sistema, como valores de pressão, temperatura e caudal.



Figura 31 - Sinóptico de gráficos

- Alarmes - histórico de todos os alarmes que ocorreram ou estão a ocorrer, com códigos de cores para identificar o estado do alarme, isto é, se já foi confirmado pelo operador, se continua a ocorrer, etc.

ALARMS HISTORY			
[Red]	08:21:48 PM	08:21:48 PM	▲ ▲ ▲ ▼ ▼
[Blue]	08:21:48 PM	08:21:48 PM	
[Blue]	08:21:48 PM	08:21:48 PM	
[Green]	08:21:48 PM	08:21:48 PM	
[Red]	08:21:48 PM	08:21:48 PM	
[Blue]	08:21:48 PM	08:21:48 PM	
[Blue]	08:21:48 PM	08:21:48 PM	
[Green]	08:21:48 PM	08:21:48 PM	
[Red]	08:21:48 PM	08:21:48 PM	
[Blue]	08:21:48 PM	08:21:48 PM	

Figura 32 - Sinóptico de alarmes

Capítulo 6

Conclusão

6.1. Comentários finais

Foi desenvolvido e testado com êxito um sistema combinado de monitorização e controlo de uma unidade industrial de PSA, com a criação de um sistema completamente operacional que satisfaz os requisitos e necessidades da empresa SysAdvance S.A.

Durante um conjunto de testes simples a que o sistema foi submetido, o mesmo reagiu a todas as ordens dadas dentro da especificação, ordens de paragem de emergência, início de operação e parametrização dos tempos de cada etapa.

O protótipo encontra-se, à data da escrita deste documento, a ser submetido a testes de qualidade de produção de oxigénio na Faculdade de Engenharia da Universidade do Porto.

Este documento é um bom caso de estudo de como o uso de um processo de projecto de engenharia na concepção de um novo sistema é uma forma de garantir boas práticas e formalizar o processo idealizado.

Demonstra-se também a importância do uso de *standards* na programação de autómatos, mais especificamente a norma IEC 61131-3. Foram identificadas algumas vantagens que este *standard* forneceu durante o desenvolvimento das aplicações, tais como:

- Desenvolvimento de programas bem estruturados;
- Fornece ferramentas para a decomposição do problema global em unidades mais pequenas e por isso mais facilmente tratáveis;
- Elevado nível de reutilização do código produzido;
- O suporte de uma linguagem de descrição de comportamentos sequenciais (SFC);
- Selecção flexível de linguagens de programação;

De destacar as vantagens do suporte do CX-Programmer na programação com SFC, que permitiu estruturar o programa de controlo com relativa celeridade e facilidade. O uso de nomes de variáveis significativos aliados a esta linguagem visualmente estruturada reduz

drasticamente a curva de aprendizagem para novos programadores que necessitem de acrescentar funcionalidades ao programa.

Também surgiram algumas desvantagens relativas ao uso de SFC na fase de desenvolvimento do software. O uso de SFC com autómatos da OMROM é bastante exigente em termos de consumo de memória de programa quando comparado com outras linguagens como *Ladder* (LD) ou *Structured Text* (ST), o que para projectos de maior dimensão implica adquirir versões de autómatos mais dispendiosas. A flexibilidade da programação em SFC é inversamente proporcional ao tamanho do código. Usando este projecto como exemplo, um aumento no número de estados de 12 para 20 ou mais tornaria a programação de todas as acções uma tarefa exaustiva. Nessa situação hipotética a melhor opção seria implementar todo o programa em LD recorrendo a endereçamento indirecto.

Espera-se que futuras versões de software implementem ferramentas que tornem a programação em SFC mais flexível e menos exigente em termos de memória.

Podemos afirmar que as relações entre o mundo empresarial e o mundo académico através de projectos inovadores são uma mais-valia bilateral responsável pela criação de produtos de elevado valor acrescentado.

Citando Woody Allen:

“If you're not failing every now and again, it's a sign you're not doing anything very innovative.”

6.2. Futuros desenvolvimentos

Algumas ideias sobre futuros desenvolvimentos e implementações são apresentadas de seguida sob a forma de tópicos.

Hardware

Relativamente ao hardware sugerem-se os seguintes desenvolvimentos:

- Substituição do actual armário metálico por outro de plástico com maiores dimensões de forma a ser possível instalar mais equipamento;
- Refazer todas as ligações no novo armário de forma a obter uma cablagem com aspecto mais apresentável;
- Colocar etiquetas em todos os cabos de forma a facilitar a sua manutenção e o despiste de avarias;

- Fazer esquemas eléctricos e pneumáticos da unidade PSA;
- Instalar um transformador 230/115VAC para alimentar o *Booster* que será ligado ao processo de PSA;
- Instalar um transdutor de corrente (LEM) e adquirir o seu sinal pelo módulo de entradas analógicas do autómato para monitorização da corrente e potência consumidas pelo sistema;
- Instalar dois sensores de temperatura na superfície dos motores de modo a monitorizar o seu funcionamento;
- Instalar dois sensores de temperatura na unidade PSA de modo a monitorizar o processo;
- Instalar quatro ventoinhas de forma a obter a melhor ventilação possível da unidade;
- Realizar um estudo exaustivo dos modos de falha e avarias possíveis do sistema, usando ferramentas como *Failure Mode Effect Analysis* (FMEA) e *Failure Mode Effect and Criticality Analysis* (FMECA).

Software

- Reprogramar o software de controlo na linguagem Ladder (LD) de forma a obter compatibilidade com mais modelos de autómatos que não suportam SFC;
- Implementar protecções para picos de corrente e/ou temperatura que coloquem o sistema num estado seguro;
- Acrescentar novas funcionalidades ao software da consola HMI;
- Desenvolver uma aplicação em Visual Basic.NET para monitorização e controlo remoto de vários processos em simultâneo.

Referências

- [1] Ralph M. Ford, Chris S. Coulston, “Design for Electrical and Computer Engineers: Theory, Concepts and Practice”, McGraw-Hill, 2008.
- [2] K. Erickson, “Programmable Logic Controllers - An Emphasis on Design and Application”, Dogwood Valley Press, LLC, 2005.
- [3] M. Groover, “Automation, Production Systems and Computer-Integrated Manufacturing”, Prentice Hall, 2008.
- [4] Lin Lin, “Numerical Simulation of Pressure Swing Adsorption Process”, Xidian University, 1990.
- [5] N. A. Downie, “Industrial Gases”, Kluwer Academic Publishers, 2002.
- [6] Douglas M. Ruthven, “Pressure Swing Adsorption”, VCH Publishers, 1994.
- [7] Heinz-Wolfgang Häring, “Industrial Gases Processing”, Wiley VCH, 2008.
- [8] “Understanding the IEC61131-3 Programming Languages”, Bosch Rexroth Corporation, 2009.
- [9] Manuais de operação e programação de equipamento OMRON. Disponível em <http://industrial.omron.pt/>. Acesso em Janeiro 2010.