

Faculdade de Engenharia da Universidade do Porto



FEUP

Integração de Sistemas de Identificação Automáticos

Nuno Miguel Silva Mota

VERSÃO FINAL

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

Orientador: Prof. Dr. Paulo Portugal
Julho de 2010

A Dissertação intitulada

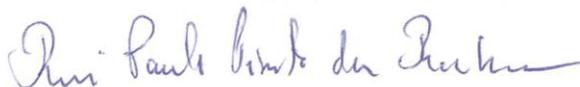
“INTEGRAÇÃO DE SISTEMAS DE IDENTIFICAÇÃO AUTOMÁTICOS”

foi aprovada em provas realizadas em 23/Julho/2010

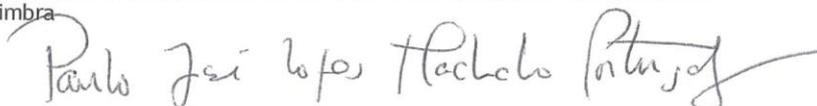
o júri



Presidente Professor Doutor Luis Miguel Pinho de Almeida
Professor Associado do Departamento de Engenharia Electrotécnica e de
Computadores da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Rui Paulo Pinto Rocha
Professor Auxiliar do Departamento de Engenharia Electrotécnica e de
Computadores da Faculdade de Ciências e Tecnologia da Universidade de
Coimbra



Professor Doutor Paulo José Lopes Machado Portugal
Professor Auxiliar do Departamento de Engenharia Electrotécnica e de
Computadores da Faculdade de Engenharia da Universidade do Porto
(Orientador);

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 - Nuno Miguel Silva Mota

Faculdade de Engenharia da Universidade do Porto

Resumo

Na presente dissertação, propõe-se uma solução para integrar três tecnologias de identificação e recolha de dados automáticos: leitores RFID, leitor código de barras e impressora de código de barras, num sistema para ensino de tecnologias de automação. O objectivo é implementar uma aplicação numa plataforma baseada num autómato programável, que disponibilize funções para ler e escrever dados a partir desses dispositivos, utilizando o protocolo de comunicação Modbus.

Pretende-se também disponibilizar ao utilizador duas interfaces para aceder à plataforma. Para tal desenvolveu-se uma interface Web, que permite aceder à plataforma remotamente utilizando um browser. Alternativamente, pode ser acedido utilizando qualquer cliente Modbus.

Abstract

In this dissertation, it was proposed a solution for integrating three technologies for the identification and automatic data collection: RFID readers, barcode reader and barcode printer, a system for teaching automation technology. The aim is to implement a platform based on a programmable controller that provides functions to read and write data from those devices using the Modbus communication protocol

Another aim is to provide to the user two interfaces to access the platform. Therefore it has developed a Web interface, which allows access to the platform remotely using a Web browser. Alternatively, it can be accessed using any client Modbus.

Agradecimentos

Gostaria de agradecer ao meu orientador, Professor Doutor Paulo Portugal o facto de ter acreditado em mim para desenvolver este trabalho, os seus conselhos e a sua preocupação.

Aos meus colegas que passaram o semestre a elaborar um trabalho semelhante de dissertação mas que, sempre que necessário, deram a sua opinião, ajuda e principalmente motivação quando o caminho se tornou por vezes mais difícil.

A todos os que estiveram envolvidos e me ajudaram no desenvolvimento deste trabalho mas não foram mencionados agradeço também profundamente.

Por último, mas em primeiro plano, gostaria de agradecer à minha família em especial aos meus pais, pelo facto de terem investido e acreditado sempre em mim, proporcionando-me a oportunidade de estar a terminar um curso superior.

*“A good scientist is a person with original ideas.
A good engineer is a person who makes a design
that works with as few original ideas as possible.
There are no prima donnas in engineering.”*

Freeman Dyson

Índice

Resumo	i
Abstract	iii
Agradecimentos	v
Índice	ix
Lista de Figura	xi
Lista de Tabelas	xv
Abreviaturas	xvii
Capítulo 1	1
Introdução	1
1.1 - Motivação da dissertação.....	1
1.2 - Objectivo da dissertação	1
1.3 - Organização do documento	2
Capítulo 2	3
Tecnologias de Identificação e Recolha de Dados Automáticos	3
2.1 - Identificação e recolha de dados automáticos	3
2.2 - Códigos de Barras	5
2.2.1 - Introdução	5
2.2.2 - Código barras linear.....	6
2.2.3 - Leitores de códigos de barras	11
2.2.4 - Código barras bidimensional	12
2.3 - Identificação por Radiofrequência - RFID	13
2.3.1 - Introdução	13
2.3.2 - Princípio de Funcionamento.....	14
2.3.3 - Frequências de comunicação.....	17
2.3.4 - Standardização de Protocolos RFID	18
2.4 - Outras Tecnologias de Identificação Automática	20
2.4.1 - Reconhecimento Óptico de Caracteres (OCR).....	20
2.4.2 - Visão Industrial	20
2.4.3 - Ecrãs tácteis.....	21
2.4.4 - Cartões inteligentes (Smart cards).....	21
2.4.5 - Tecnologias Biométricas	23

2.5 - Sumário das Tecnologias de Identificação e Recolha de Dados Automáticos	23
Capítulo 3	25
Descrição do Problema	25
3.1 - Linha de Fabrico Flexível.....	25
3.2 - Análise de requisitos	27
3.3 - Leitor RFID	28
3.4 - Leitor de Código de barras.....	33
3.5 - Impressora	35
3.6 - Plataforma de Integração	37
3.7 - Protocolo de comunicação Modbus	38
3.8 - Servidor Modbus no Autómato Programável	41
Capítulo 4	43
Proposta da Aplicação	43
4.1 - Introdução	43
4.2 - Integração Leitores RFID	48
4.2.1 - Introdução	48
4.2.2 - Funções Implementadas na Aplicação	50
4.2.3 - Meio Físico de Comunicação.....	57
4.3 - Integração do Leitor de Código de Barras	58
4.3.1 - Introdução	58
4.3.2 - Função implementada na Aplicação	58
4.3.3 - Meio Físico de Comunicação.....	58
4.4 - Integração da Impressora	59
4.4.1 - Introdução	59
4.4.2 - Funções Implementadas na Aplicação	60
4.4.1 - Meio Físico de Comunicação.....	64
4.5 - Funções do Autómato Programável.....	64
4.6 - Interface Web	65
4.6.1 - Objectivo	65
4.6.2 - Desenvolvimento da Aplicação.....	65
Capítulo 5	71
Validação da Solução Proposta.....	71
5.1 - Software Modbus Poll	71
5.2 - Interface Web	75
Capítulo 6	83
Conclusões e Futuros Desenvolvimentos.....	83
6.1 - Conclusões gerais	83
6.2 - Futuros Desenvolvimentos	84
Referências	85

Lista de Figura

Figura 2.1 - Código de barras bidimensional (à direita) e linear (à esquerda).....	6
Figura 2.2 - Código de barras modulado por largura.	6
Figura 2.3 - Código de barras modulado em altura.	6
Figura 2.4 - <i>Narrow bar</i> [5].	7
Figura 2.5 - <i>Quiet zone</i> ou “ <i>no print</i> ” <i>zone</i> [5].....	7
Figura 2.6 - <i>Start</i> e <i>Stop Character</i> [5].	8
Figura 2.7 - <i>Data Characters</i> e <i>Check Character</i> [5].....	8
Figura 2.8 - <i>Intercharacter Gap</i> [5].....	9
Figura 2.9 - Simbologia EAN-13.....	9
Figura 2.10 - Simbologia <i>Code 128</i>	10
Figura 2.11 - Simbologia <i>Code 39</i>	10
Figura 2.12 - Simbologia <i>Interleaved 2 of 5</i>	10
Figura 2.13 - Princípio de funcionamento do leitor de código de barras [5].	11
Figura 2.14 - Leitor de código de barras fixo (à esquerda) e portátil (à direita).....	12
Figura 2.15 - Simbologias de códigos de barras 2-D do tipo <i>stacked</i>	13
Figura 2.16 - Simbologias de código de barras 2-D tipo <i>Matrix</i>	13
Figura 2.17 - Cronograma da tecnologia RFID [7].	14
Figura 2.18 - Sistema básico RFID [7].	15
Figura 2.19 - Exemplo leitor RFID [7].	15
Figura 2.20 - Exemplos de <i>transponder</i>	16
Figura 2.21 - Princípio de funcionamento do RFID.....	17
Figura 2.22 - Exemplos de OCR.....	20

Figura 2.23 - Sistemas de visão industrial.	20
Figura 2.24 - Exemplo de ecrãs tácteis.	21
Figura 2.25 - Contactos IC.....	22
Figura 2.26 - Exemplos de cartões inteligentes.	22
Figura 2.27 - Exemplo de leitores de cartões inteligentes.	22
Figura 2.28 - Exemplo de tecnologias biométricas.	23
Figura 3.1 - Linha de fabrico flexível [5].	26
Figura 3.2 - Partes constituintes da linha de fabrico flexível [5].	26
Figura 3.3 - Arquitectura do sistema com integração dos dispositivos AIDC.	28
Figura 3.4 - Leitor RFID modelo ID ISC.PR101-A.	29
Figura 3.5 - Troca de dados entre o host e o leitor RFID utilizando os modos <i>Configuration Commands</i> e <i>Control Commands</i> [11].	29
Figura 3.6 - Troca de dados entre host e o leitor RFID utilizando os modos <i>ISO15693 Host Comands: Addressed Mode</i> [11].	31
Figura 3.7 - Troca de dados entre host e o leitor RFID utilizando os modos <i>ISO15693 Host Comands: Non-Addressed Mode</i> [11].	31
Figura 3.8 - Troca de dados entre host e o leitor RFID utilizando os modos <i>ISO15693 Host Comands: Selected Mode</i> [11].	32
Figura 3.9 - Peça da linha de fabrico flexível com <i>Tag-it HF-I Plus</i> embutida [5].	33
Figura 3.10 - Scanner manual CCD 1021Plus.	34
Figura 3.11 - Impressora <i>Able Systems Limited</i> modelo Ap863-F.	36
Figura 3.12 - Autómato <i>Modicon TSX Premium</i>	38
Figura 3.13 - Construção da trama TCP/IP- <i>ethernet</i>	39
Figura 3.14 - Trama Modbus/TCP.	39
Figura 3.15 - Diferenças entre trama Modbus e Modbus/TCP.....	40
Figura 3.16 - Relação entre registos Modbus e posições de memória do autómato.	41
Figura 4.1 - Fluxograma que descreve o princípio de funcionamento da aplicação.	48
Figura 4.2 - Estrutura das tramas do protocolo <i>Standart Protocol</i> [11].....	49
Figura 4.3 - Estrutura das tramas do protocolo <i>Advanced Protocol</i> [11].	49
Figura 4.4 - Fluxograma que descreve funcionamento do sistema RFID.	50
Figura 4.5 - Fluxograma que descreve funcionamento do leitor RFID com endereço 0x01.....	51
Figura 4.6 - Fluxograma da função <i>Inventory</i>	52

Figura 4.7 - Fluxograma da função <i>Write Multiple Blocks</i>	53
Figura 4.8 - Fluxograma da função <i>Read Multiple Blocks</i>	54
Figura 4.9 - Trama de envio e resposta do comando <i>Inventory</i> [11].	55
Figura 4.10 - Trama de envio e resposta do comando <i>Read Multiple Blocks</i> [11].....	56
Figura 4.11 - Trama de envio e resposta do comando <i>Write Multiple Blocks</i> [11].....	56
Figura 4.12 - Trama de envio do comando CPU Reset [11].....	56
Figura 4.13 - Integração dos leitores RFID no autómato.	57
Figura 4.14 - Esquema de ligações do leitor RFID no autómato.	57
Figura 4.15 - Integração do leitor código de barras no autómato.....	59
Figura 4.16 - Esquema de ligação do <i>scanner</i> no autómato.	59
Figura 4.17 - Código de barras impresso.....	60
Figura 4.18 - Fluxograma que descreve aplicação da impressora.	61
Figura 4.19 - Fluxograma da função imprimir código de barras.	61
Figura 4.20 - Fluxograma da função imprimir texto.	62
Figura 4.21 - Fluxograma da função imprimir imagem.	62
Figura 4.22 - Fluxograma da função imprimir comandos especiais.	63
Figura 4.23 - Integração da impressora no autómato.	64
Figura 4.24 - Esquema de ligações da impressora no autómato.	64
Figura 4.25 - Interfaces gráficas dos menus Consultar dados RFID.	66
Figura 4.26 - Interface gráfica do menu Comandos RFID.	67
Figura 4.27 - Interface gráfica do menu Código de Barras.	67
Figura 4.28 - Interface gráfica do menu Impressora.....	68
Figura 4.29 - Interface gráfica do menu Login.....	68
Figura 4.30 - Visão da interface Web utilizando casos de uso.	69
Figura 5.1 - Interface gráfica com o utilizador do <i>Software Modbus Poll</i>	71
Figura 5.2 - Interface gráfica dos comandos enviados para os leitores RFID.....	72
Figura 5.3 - Interface gráfica com dados do leitor 2.	72
Figura 5.4 - Interface gráfica com dados do leitor 3.	73
Figura 5.5 - Interface gráfica com dados do leitor 4.	73
Figura 5.6 - Interface gráfica com dados do leitor código de barras.	73

Figura 5.7 - Interface gráfica de impressão código de barras.	74
Figura 5.8 - Interface gráfica de impressão texto.	74
Figura 5.9 - Interface gráfica de impressão imagem.	74
Figura 5.10 - Interface gráfica da aplicação Web, com utilizador não autenticado.	75
Figura 5.11 - Menu com utilizador autenticado.	75
Figura 5.12 - Interface gráfica dos comandos dos leitores RFID.....	76
Figura 5.13 - Interfaces gráficas da validação dos formulários.....	77
Figura 5.14 - Interface gráfica com escolha dos leitores RFID para consulta de dados e com validação do formulário.....	77
Figura 5.15 - Interface gráfica da consulta de dados dos leitores RFID.....	78
Figura 5.16 - Interface gráfica de consulta da leitura realizada pelo leitor de código de barras.	79
Figura 5.17 - Interface gráfica das funcionalidades disponíveis na impressora com visualização código de barras a imprimir.	79
Figura 5.18 - Interface gráfica com impressão de texto.	80
Figura 5.19 - Interface gráfica das funcionalidades da impressora, com indicação de ocorrência de erros.	80

Lista de Tabelas

Tabela 2.1 - Classificação do Sistemas RFID relativamente à frequência de comunicação [9].	18
Tabela 2.2 - Normas publicadas pela ISO, aplicáveis aos sistemas RFID [14].	19
Tabela 2.3 - Sumário das diferentes tecnologias de identificação e recolha de dados automáticos [15].	24
Tabela 3.1 - Características do leitor RFID.....	29
Tabela 3.2 - Funções disponíveis no modo de configuração.	30
Tabela 3.3 - Funções disponíveis no modo de controlo.	30
Tabela 3.4 - Funções disponíveis no modo ISO15693 Host Comands.....	32
Tabela 3.5 - Funções do protocolo de comunicação Modbus/TCP.	41
Tabela 4.1 - Organização dos registos Modbus para os leitores RFID.	44
Tabela 4.2 - Organização dos registos Modbus para o leitor de código de barras.....	45
Tabela 4.3 - Organização dos registos Modbus para impressora.....	46
Tabela 4.4 - Reorganização automática dos registos Modbus para os todos os dispositivos....	47
Tabela 4.5 - Descrição da trama do protocolo de comunicação do leitor RFID.....	49
Tabela 4.6 - Descrição da trama Inventory.	55
Tabela 4.7 - Descrição das tramas Read Multiple Blocks e Write Multiple Blocks.....	56

Abreviaturas

Lista de abreviaturas (ordenadas por ordem alfabética)

ADC	Automatic Data Capture
AIAG	Automotive Industry Action Group
AIDC	Automatic Identification and Data Capture
AND	Ácido desoxirribonucleico
ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
Auto-ID	Automatic Identification
Bps	Bytes por segundo
CC	Corrente Contínua
CCD	Charge-Coupled Device
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
DEEC	Departamento de Engenharia Electrotécnica e de Computadores
EAN	European Article Numbering
EEPROM	Electrically-Erasable Programmable Read-Only Memory
EPC	Electronic Product Code
EPROM	Erasable Programmable Read-only Memory
GHz	Gigahertz
GND	Ground
HF	High Frequency
HIBC	Health Industry Bar Code Council
HTML	HyperText Markup Language
IC	Integrated Circuit
IEC	International Electrotechnical Commission
IP	Internet Protocol
ISO	International for Standardization

KHz	Kilohertz
LED	Light-Emitting Diode
LF	Low Frequency
LSB	Least Significant Bit
MAC	Media Access Control
MBAP	Modbus Application Protocol
MHz	Megahertz
MIT	Massachusetts Institute of Technology
MSB	Most Significant Bit
OCR	Optical Character Recognition
PDU	Protocol Data Units
PHP	Hypertext Preprocessor
PLC	Programmable Control System
PS/2	Personal System/2
RAM	Random-access Memory
RF	Radio Frequency
RFID	Radio Frequency Identification
RS-232	Recommended Standard 232
RS-485	Recommended Standard 485
RxD	Receive Data
SG	Signal Ground
SHF	Super High Frequency
SPI	Serial Peripheral Interface
TCP	Transmission Control Protocol
TxD	Transmit Data
UDP	User Datagram Protocol
UHF	Ultra High Frequency
UID	Unique Identifier
UML	Unified Modeling Language
UPC	Universal Product Code
Web	World Wide Web

Capítulo 1

Introdução

1.1 - Motivação da dissertação

No Departamento de Engenharia Electrotécnica de Computadores (DEEC), existe um simulador (Kit) de uma linha de fabrico flexível que tem como principal objectivo, disponibilizar aos alunos um meio de aprendizagem de tecnologias de automação, utilizando um sistema industrial muito próximo da realidade. Existe a necessidade de integrar nesta linha de fabrico, um conjunto de tecnologias de identificação e recolha de dados automáticos, para dar a conhecer aos alunos novas tecnologias de automação, bem como disponibilizar uma interface para a utilização das mesmas de forma transparente. O âmbito da dissertação, parte da necessidade da integração de um conjunto de tecnologias de identificação e recolha de dados automáticos, para utilização dos alunos na linha de fabrico flexível.

1.2 - Objectivo da dissertação

Pretende-se com a realização da dissertação, a integração de um conjunto de tecnologias de identificação e recolha de dados automáticos numa mesma aplicação baseada num autómato, sem a necessidade de interagir directamente com cada um dos dispositivos. O protocolo de comunicação comum pretendido é o Modbus. Assim, o objectivo da dissertação é criar uma aplicação no autómato, capaz de comunicar com os diferentes dispositivos e permitir realizar algumas funções, tais como, leitura e escrita de dados. Pretende-se ainda aceder a estes dados bem como enviar comandos de forma transparente, utilizando uma interface Web ou a partir de qualquer software que suporte o protocolo de comunicação Modbus.

1.3 - Organização do documento

Este documento está organizado em 6 capítulos onde se descreve todo o trabalho realizado, cujos conteúdos são descritos, de forma sucinta, na presente secção.

O primeiro capítulo é o capítulo introdutório e serve para explicar a motivação, a necessidade e os objectivos pretendidos com a realização desta dissertação.

No capítulo 2, faz-se uma introdução às tecnologias de identificação e recolha de dados automáticos, apresentando algumas tecnologias e princípio de funcionamento. As tecnologias a utilizar na realização do projecto merecerão especial atenção.

No capítulo 3, descreve-se o problema, apresenta-se a análise de requisitos, arquitecturas adoptadas, os equipamentos utilizados na realização do projecto e o protocolo de comunicação comum pretendido.

No capítulo 4, apresenta-se o trabalho desenvolvido, desde o estudo de protocolos de comunicação, meios físicos e implementação das diferentes funções na aplicação.

No capítulo 5, são descritos os testes realizados para a validação da solução proposta.

Por fim, no capítulo 6, apresentam-se as conclusões da dissertação realizada e propostas para desenvolvimentos futuros.

Capítulo 2

Tecnologias de Identificação e Recolha de Dados Automáticos

2.1 - Identificação e recolha de dados automáticos

Os sistemas de identificação e recolha de dados automáticos (AIDC - Automatic Identification and Data Capture) ou também designados apenas por sistemas de identificação automática (Auto-ID automatic identification) ou ainda sistemas de recolha de dados automáticos (ADC - Automatic data capture), são tecnologias utilizadas para adicionar dados directamente num computador, PLC ou microprocessador. Permitem identificar, monitorizar e recolher dados de objectos ou pessoas com elevada fiabilidade, sem intervenção humana e codificar uma vasta gama de informação, desde a mais simples até informação mais detalhada. Estas tecnologias estão presentes no nosso dia-a-dia, desde o código de barras dos produtos dos hipermercados, aos RFID utilizados na Via Verde das portagens, os cartões de banda magnética, a impressão digital utilizada para autenticação em computadores portáteis, o reconhecimento por voz na marcação de números de telemóvel, são alguns exemplos entre muitos. Na indústria a sua utilização ganhou uma enorme importância, tanto na produção como na movimentação de materiais. Normalmente a movimentação dos materiais, envolve acções de expedição, recepção armazenamento e movimentação de objectos e ordens de produção, tais como, produtividade dos funcionários, desempenho do processo produtivo e monitorização da taxa de utilização das máquinas. Outras áreas de aplicação destas tecnologias são sistemas de segurança, distribuição e armazenamento, identificação de pacientes nos hospitais, entre outros.

Existe uma grande variedade de soluções que estas tecnologias oferecem, não havendo uma solução óptima. Cada tecnologia tem as suas vantagens específicas e características, sendo a melhor solução para uma determinada aplicação mas não sendo para outra.

Estes sistemas iniciaram os seus primeiros passos nos anos setenta [1], tendo substituído a recolha de dados manual. Normalmente este método implica a intervenção humana, para a

recolha dos dados para papel para posteriormente os mesmos serem inseridos num computador recorrendo ao teclado. Este método apresenta desvantagens significativas tais como, mão-de-obra, tempo e ocorrência de erros com maior facilidade [2].

Com a utilização das tecnologias de identificação e recolha de dados automática, estas desvantagens são parcialmente eliminadas, obtendo-se maior fiabilidade e segurança na recolha dos dados.

Os sistemas de identificação e recolha de dados automáticos podem ser classificados em seis categorias, sendo elas:

- Tecnologia electromagnética: corresponde à identificação por radiofrequência. Constituído por três componentes, antena, *transceiver* e *transponder*.
- Tecnologia óptica: utiliza um contraste de símbolos gráficos para codificar informação, que são interpretados por um *scanner* óptico. Esta tecnologia inclui códigos de barras unidimensionais e bidimensionais, visão industrial e reconhecimento óptico de caracteres (OCR).
- Tecnologia magnética: corresponde a codificar a informação sob a forma magnética. O exemplo mais comum, é o cartão magnético, que permite armazenar qualquer tipo de dados digitais.
- Tecnologia de *touch screen*: funciona por acção de um simples toque.
- Tecnologia de biometria: Caracteriza-se pela identificação de pessoas através do reconhecimento de características humanas únicas tais como, análise de impressão digital, reconhecimento fácil, de retina e voz.
- Cartões inteligentes: São cartões que possuem um circuito integrado, com a capacidade de armazenar e processar dados.

Algumas das aplicações de sistemas de identificação e recolha de dados automáticos, requerem intervenção humana, para o procedimento de recolha de dados, normalmente para manusear o dispositivo. Na verdade, estes tipos de aplicações não são automáticos, mas sim semi-automáticos. As tecnologias mencionadas anteriormente podem ter os dois tipos de aplicações.

Habitualmente, todas as tecnologias são constituídas por três dispositivos, um codificador de dados, um leitor ou *scanner* e um descodificador de dados. O primeiro guarda os dados sobre a forma de símbolos ou sinais que representam caracteres alfanuméricos, como por exemplo o código de barras. O leitor ou *scanner* lê os dados codificados e transforma os mesmos num sinal eléctrico. Por fim o descodificador, que transforma o sinal eléctrico novamente em caracteres alfanuméricos. Estes dois últimos, normalmente estão integrados apenas num dispositivo, como por exemplo o leitor de código de barras.

2.2 - Códigos de Barras

2.2.1 - Introdução

Os códigos de barras surgiram no comércio e na indústria alimentar no início dos anos 70 [1], passando desde então a ser utilizados em larga escala. Antes destes apenas existia a identificação manual. Os rótulos eram escritos à mão ou anexados em papel químico aos objectos que necessitavam de ser identificados. Os códigos de barras representam uma forma fácil, rápida e precisa para a recolha de dados. Permitem a identificação de uma forma eficiente e precisa de produtos, não sendo possível obter utilizando um sistema manual. As principais vantagens estão no automatismo das operações que ocorre instantaneamente como se fosse uma transacção, normalmente referido como recolha de dados em tempo real. Estudos provaram que a taxa de leituras de entrada de dados, quando utilizado um sistema de recolha de dados automático e tecnologia de código de barras, é aproximadamente de um erro num milhão de caracteres, ao contrário de um erro em cada trezentos quando a recolha de dados é manual [3]. A explosão dos computadores no início dos anos 80, serviu para difundir e promover a utilização de aplicações de códigos de barras para a recolha de dados de forma automática. Passados alguns anos, os códigos de barras passaram a ser uma necessidade virtual na recolha rápida e no processamento de dados, e estão presentes em qualquer actividade. Os códigos de barras podem ser utilizados em todas as áreas, comerciais ou não, tais como por exemplo:

- Identificação e registo de inventário
- Registo de entrada e saída de mercadorias numa fábrica
- Controlar a produção
- Registar a entrada e saída de correio
- Identificação de documentos
- Identificação de pessoas
- Etc.

Para melhor compreensão, os códigos de barras podem ser vistos como “código binário”, codificados sob a forma de barras. Obedecem a normas, desenvolvidas e aperfeiçoadas ao longo dos anos, constituído um conjunto de representações, designado simbologias. A regulamentação na utilização e a normalização dos códigos de barras é da responsabilidade de várias associações internacionais, como por exemplo, ANSI - *American National Standards Institute*, HIBC - *Health Industry Bar Code Council* e a AIAG - *Automotive Industry Action Group*.

Existem dois tipos de códigos de barras, unidimensional ou linear e bidimensional. O código unidimensional ou linear, é constituído por uma sequência de barras e apenas pode ser lido num sentido, o código bidimensional é constituído por uma matriz de barras e necessita de ser lido em duas direcções.



Figura 2.1 - Código de barras bidimensional (à direita) e linear (à esquerda).

2.2.2 - Código barras linear

O código de barras linear é o mais utilizado, entre as tecnologias de identificação e recolha de dados automática. Pode codificar a informação de dois modos diferentes, modulação por altura (*Height-Modulated*) ou por largura (*Width-Modulated*). Na modulação por largura (Figura 2.2), os dados são codificados numa sequência de barras e espaços de largura variável. Na modulação por altura (Figura 2.3), a modelação é feita de igual forma por uma sequência de barras espaçadas, mas de altura variável.



Figura 2.2 - Código de barras modulado por largura.



Figura 2.3 - Código de barras modulado em altura.

O código de barras modelado em largura, é claramente o mais utilizado, o código de barras modelado em altura não possui ainda grandes aplicações. Por esta razão, apenas se apresentará o código de barras modelado em largura.

Como anteriormente referido, o código de barras modulado em largura consiste numa sequência de barras estreitas e largas intercaladas por espaços, também eles estreitos e largos.

Habitualmente as barras apresentam uma cor preta ou escura, e os espaços uma cor branca ou clara, criando um contraste, o que facilita a leitura. Existem diversas simbologias, onde as barras têm diferentes dimensões, bem como diferentes codificações. Algumas simbologias surgiram por razões históricas ou políticas, enquanto outras foram estudadas e criadas tendo em conta as suas vantagens e aplicações. De seguida serão apresentadas as características comuns à maioria dos diferentes tipos de códigos de barras.

- *Narrow bar* ou *x-dimension*: Define a dimensão nominal das barras ou dos espaços mais estreitos de cada simbologia de código de barras, e é utilizado para determinar o tamanho da escala do código de barras. A unidade de medida normalmente é o milímetro ou nos Estados Unidos o mil (*0.001 inch*). Deve utilizar-se o *narrow bar* maior que aplicação permite, pois será mais fácil tanto imprimir como ler o código de barras. Além disso diminui os possíveis erros de leitura, dado que a área de leitura é maior. A densidade do código de barras, corresponde ao número de caracteres por milímetro ou *inch*, e será tanto maior quanto menor for a dimensão do *narrow bar*.



Figura 2.4 - *Narrow bar* [5].

- *Quiet zone*: O tamanho do código de barras não se limita à zona onde começa e termina as barras. O *quiet zone* ou "*no print*" zone, corresponde pelo menos a dez vezes a largura do *narrow bar*, e encontra-se no início e fim do código. Estas zonas são importantes, dado que se não forem visíveis ou contiverem texto ou grafismo, o código pode não ser lido.

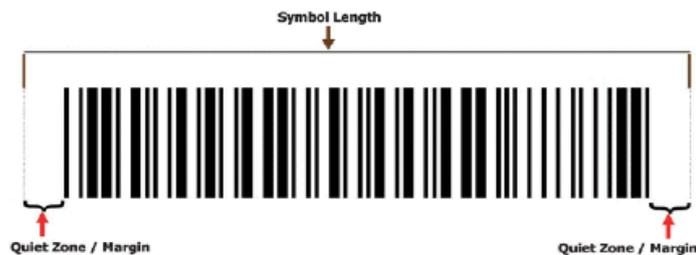


Figura 2.5 - *Quiet zone* ou "*no print*" zone [5].

- *Start/Stop Character*: Os leitores de códigos de barras podem ler nos dois sentidos, ao contrário das pessoas que apenas lêem num sentido (da esquerda para a direita). Isto leva a que a maioria das simbologias utilize um carácter de início e de paragem, para indicar ao leitor a ordem dos dados codificados. No entanto, nem todas as simbologias utilizam estes caracteres, por exemplo *Pharma Code*, e por isso depende da direcção em que o código barras é colocado, podendo dar origem a leituras diferentes.



Figura 2.6 - *Start e Stop Character* [5].

- *Data Characters e Check Character*: A informação do código de barras é codificada nos caracteres de dados (*Data Characters*), e o seu tamanho depende da simbologia utilizada. São estes dados que o leitor irá ler e apresentar ao utilizador. Também dependente da simbologia utilizada, o carácter de confirmação (*Check Character*) pode ser incluído no código de barras, utilizado para detecção posterior de erros. Em alguns casos, este carácter é incluído automaticamente, noutros pode ser opcional. Sempre que possível, deve ser utilizado, ainda mais quando se pretende integridade na aplicação.



Figura 2.7 - *Data Characters e Check Character* [5].

- *Intercharacter Gap*: As simbologias de códigos de barras lineares podem ainda ser classificadas como discretas ou contínuas. No caso contínuo, um carácter começa imediatamente a seguir ao anterior, sem que haja intervalo entre eles. Um carácter tipicamente começa numa barra e termina num espaço e todos os espaços fazem parte do código. Nos códigos de barras discretos, existem intervalos entre caracteres (*Intercharacter Gap*), que não contém dados.



Figura 2.8 - Intercharacter Gap[5].

Para além das características apresentadas anteriormente, existem outras características importantes de um código de barras, a altura das barras e o rácio de contraste. Relativamente à altura, é fácil perceber que quanto maior a altura das barras mais fácil será a leitura, aumentado a possibilidade de fazer leituras correctas. Quanto ao rácio de contraste, é a percentagem de contraste entre as barras e os espaços como já foi referido, facilita a descodificação do *scanner* e evita erros. Algumas das diversas simbologias serão apresentadas de seguida [3].

- **UPC/EAN**

A simbologia UPC/EAN (Figura 2.9) foi a primeira a ser largamente utilizada para a identificação de produtos. O seu sucesso levou à criação de um código similar, o EAN. Existem actualmente cinco versões do UPC, sendo as mais comuns de A e E, e na EAN, as EAN-8 e EAN-13. Todas as simbologias são numéricas. Importante salientar que as simbologias são contínuas, cada carácter é codificado por 2 barras e 2 espaços.



Figura 2.9 - Simbologia EAN-13.

- **Code 128**

O *code 128* (Figura 2.10), é uma simbologia alfanumérica com tamanho variável. Apresenta uma grande vantagem relativamente a todas as simbologias lineares, a alta densidade de codificação, isto é, utiliza uma menor área para codificar 6 caracteres ou mais. Permite representar todos os caracteres ASCII, usa 3 espaços e 3 barras para codificar cada dígito.



Figura 2.10 - Simbologia Code 128.

- **Code 39**

Este código também conhecido por *Code 3 of 9* (Figura 2.11), é o código mais utilizado na maioria das aplicações. A sua larga aceitação deve-se ao facto de ter sido a primeira simbologia alfanumérica a ser desenvolvida, e poder ser lida com a maioria dos leitores de códigos de barras utilizando a sua configuração por defeito. É o mais antigo dos códigos de barras actuais. Tem comprimento variável, e pode suportar qualquer número de caracteres desde que aceite pelo leitor. Pode representar 26 letras maiúsculas, 10 dígitos (0-9) e 7 caracteres especiais. Cada carácter é representado por 5 barras e 4 espaços, em que 3 destes elementos são sempre largos e o carácter de início e de paragem é um asterisco.



Figura 2.11 - Simbologia Code 39.

- **Interleaved 2 of 5**

A simbologia *Interleaved 2 of 5* (Figura 2.12), é utilizada para codificar apenas números com tamanho variável. Contém *Check character*, mas é opcional. Os dígitos são codificados por 5 barras, 2 delas largas e 5 espaços, 2 deles largos. Um dígito está codificado numa barra e o seguinte num espaço alternadamente, daí o nome *Interleaved*, intercalado. Deduz-se facilmente, que esta é uma simbologia contínua pois não existem espaços sem informação. Os caracteres de início e de paragem são formados pela sequência *narrow bar*, *narrow space*, *narrow bar*, *narrow space* e *wide bar*, *narrow space* e *narrow bar*.



Figura 2.12 - Simbologia Interleaved 2 of 5.

2.2.3 - Leitores de códigos de barras

A função principal do leitor de código de barras ou *scanner*, é ler a imagem representada pelo código de barras. Na sua forma mais básica, um *scanner* lê e mede a ausência e presença de luz, isto é barras pretas ou escuras, espaços brancos ou claros, e converte essa informação em sinais eléctricos, que por sua vez podem ser convertidos em dados reconhecidos pelo computador.

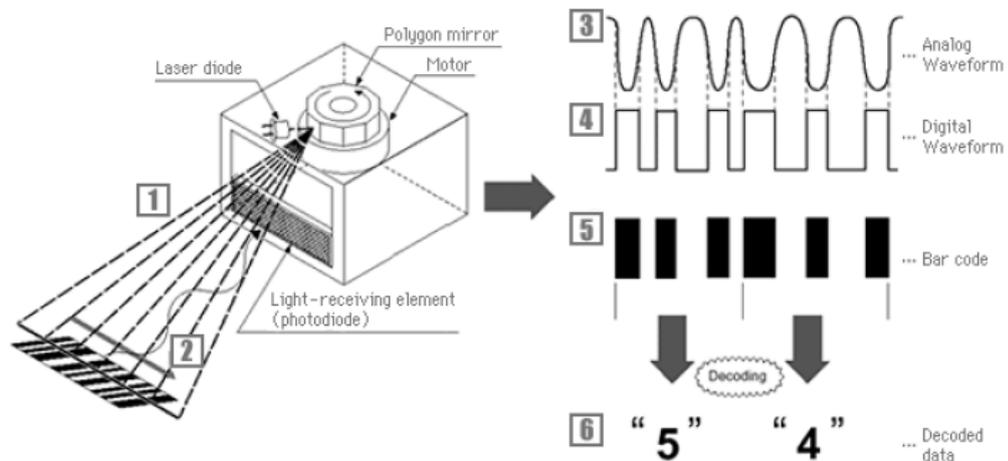


Figura 2.13 - Princípio de funcionamento do leitor de código de barras [5].

Os leitores de códigos de barras podem ser totalmente automáticos, ou portáteis necessitando de intervenção humana. Uma das classificações está associada quanto ao tipo de feixe de luz, que pode ser fixo ou móvel. Para que a leitura seja bem sucedida, o feixe de luz deve passar por cima do código de barras a uma velocidade aproximadamente constante. Em alguns equipamentos o feixe de luz é estacionário, a leitura é feita pelo movimento da mão do operador ou deslocando o código de barras na zona incidência do feixe de luz. Noutros equipamentos (*scanner laser*), o feixe de luz oscila permitindo a leitura dos códigos de barras em pequenos intervalos de tempo. Em todas estas situações, o operador tem a necessidade de apontar o feixe de luz para o código de barras para que seja possível a sua leitura.

A maioria dos leitores é portátil. Numa biblioteca a mobilidade exigida na leitura de um código de barras num documento ou cartão de leitor, exige que este tipo de leitor seja o mais adequado. Outros são fixos e colocados em posições não deslocáveis, normalmente colocados numa zona de leitura. Este é o tipo de leitores utilizado em lojas comerciais, como por exemplo nos hipermercados.



Figura 2.14 - Leitor de código de barras fixo (à esquerda) e portátil (à direita).

Ainda relativamente à classificação dos leitores estes podem ser de contacto ou sem contacto. Os leitores sem contacto são o tipo mais vulgar na recolha de dados. Os equipamentos de leitura à distância emitem um feixe de luz que é apontado para o código de barras. Neste caso não existe qualquer contacto do equipamento com a etiqueta. Esta é a melhor forma de preservar a etiqueta.

Nos leitores de contacto, a leitura é feita por contacto do equipamento de leitura com o código de barras. Este tipo de leitor, pelas suas características e frequência de utilização pode causar danos no código de barras, e como consequência erros de leitura. Existem quatro tipos de leitores de códigos de barras, leitores do tipo esférica, *scanner*, *laser*, leitores de CCD e leitores com câmaras, cada um usa uma tecnologia diferente para ler e decodificar um código de barras. No entanto, todas resultam num sinal digital que deve ser convertido em dados reconhecidos pelo computador.

2.2.4 - Código barras bidimensional

O código de barras bidimensional surgiu em 1987. É utilizado em aplicações de marketing, soluções de autenticação e segurança, informações e até interação entre pessoas. As suas grandes vantagens relativamente aos códigos de barras lineares, é a capacidade de codificar um maior número de dados, através de uma maior densidade e poder de leitura muito superior. No entanto a sua descodificação é mais custosa, são necessários leitores de códigos de barras especiais para a leitura deste tipo de código.

Como anteriormente mencionado, contém dados em duas direcções, na vertical e horizontal, e são códigos de barras lineares empilhados uns em cima dos outros com orientação e localização. Os códigos de barras podem ser divididos em duas categorias, *matrix* e *stacked*. Os códigos *stacked* consistem em multi-linhas de códigos de barras lineares agrupados verticalmente, por exemplo, *Code 49*, *Code 16K* e *PDF417* (Figura 2.15). Os códigos

matrix (Figura 2.16) oferecem maior densidade de dados que os códigos *stacked*, contendo por isso maior informação.

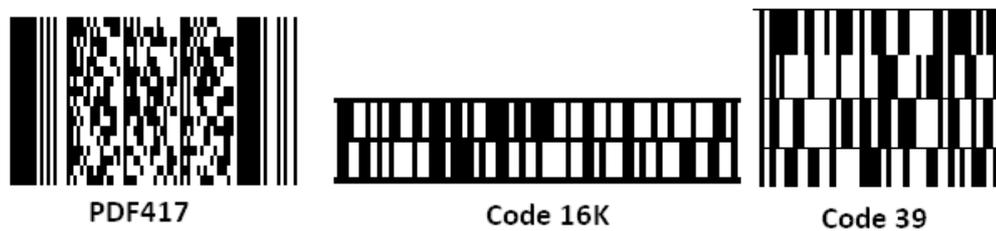


Figura 2.15 - Simbologias de códigos de barras 2-D do tipo stacked.

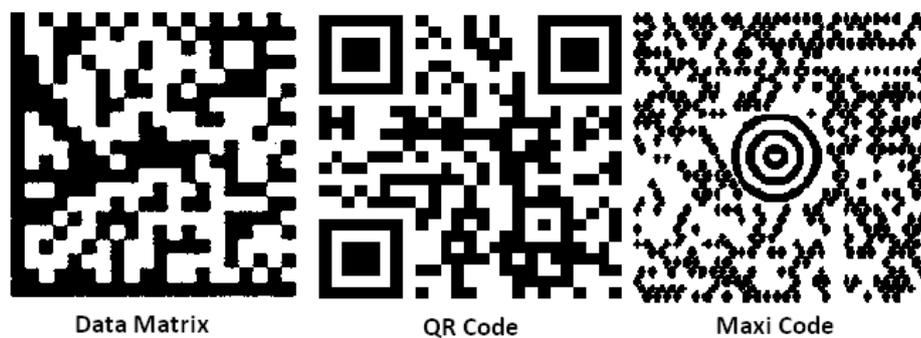


Figura 2.16 - Simbologias de código de barras 2-D tipo Matrix.

2.3 - Identificação por Radiofrequência - RFID

2.3.1 - Introdução

RFID, é um acrónimo do nome “*Radio Frequency Identification*” que em português significa, Identificação por Radiofrequência. Trata-se de um método de identificação automática através de sinais de rádio [4]. Esta tecnologia teve as suas raízes nos sistemas de radares utilizados na segunda guerra mundial. Os *Transponders*, eram utilizados nos aviões para detectar se era um inimigo ou não. A tecnologia RFID surge por junção de outras duas, o radar e as ondas de rádio (RF). Continuou a sofrer desenvolvimentos, e em 1973 surge a primeira patente com um *transponder* activo com memória regravável [6]. A sua comercialização inicia-se nos anos 80, sendo a sua primeira aplicação na identificação de gado.

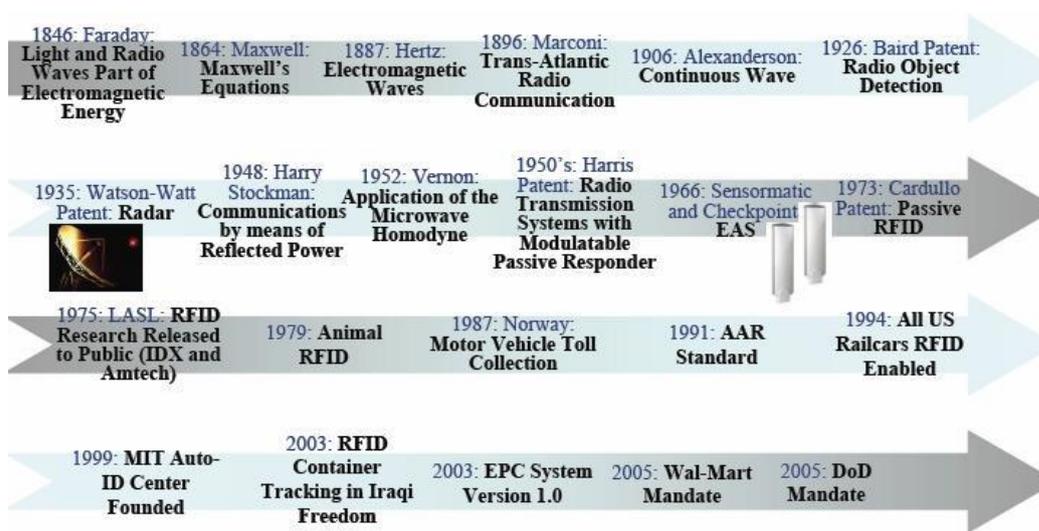


Figura 2.17 - Cronograma da tecnologia RFID [7].

Devido à sua constante desenvolvimento, a tecnologia é aplicada em variadíssimas áreas, e está cada vez mais presente no nosso dia-a-dia, seja no controlo dos produtos de hipermercados ou lojas, nos cartões para pagamentos de serviços, sistemas anti-furto, etc.

A grande dúvida nos dias de hoje passa por saber se o RFID acabará por ser a sentença de morte do já antigo código de barras. As opiniões dividem-se, embora exista uma vantagem da tecnologia RFID. Entre as mais-valias existentes, destaca-se o facto de as etiquetas não terem de estar à vista do leitor para serem lidas, podendo ser lidas em conjunto de forma praticamente simultânea. Notar ainda que estas mesmas etiquetas podem conter mais dados, que a leitura pode ser completamente automática e sem intervenção humana, e que a fiabilidade dos dados é muito elevada e que existe a possibilidade de se identificarem itens individuais, ao contrário do código de barras, que apenas permite identificar classes de objectos [7].

2.3.2 - Princípio de Funcionamento

A tecnologia RFID, representa uma tecnologia de armazenamento, leitura, gravação e manipulação de dados remotos através de comunicação de dados por radiofrequência. Uma fonte designada por leitor emite uma onda electromagnética, podendo ter diferentes comprimentos de onda. Um outro componente, designado de *transponder*, *tag* ou etiqueta recebe esse mesmo sinal e responde emitindo um sinal que por sua vez será identificado pela fonte emissora, que geralmente se encontra conectado a um computador. Na Figura 2.18, ilustra-se o princípio de funcionamento de um sistema RFID.

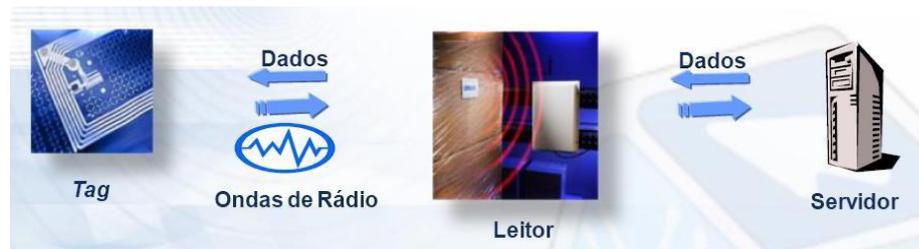


Figura 2.18 - Sistema básico RFID [7].

De uma forma geral, o sistema RFID é constituído por dois componentes, *transponder* e leitor RFID, e ambos possuem antenas que são utilizadas para comunicarem entre si.

O leitor geralmente contém um emissor e um receptor, e tanto pode ser móvel ou fixo, e é o componente de comunicação entre o sistema RFID e os sistemas externos de processamento de informações. Realiza a codificação e decodificação dos sinais emitidos/recebidos, além de ser responsável por tratar os dados recebidos. Os leitores RFID variam muito na sua complexidade, dependendo do tipo de etiqueta e das funções a serem aplicadas. Muitos dos leitores não possuem antena incorporada, actualmente os leitores mais recentes já possuem antenas incorporadas, o que incrementa o raio de acção destes. Os mais sofisticados apresentam funções de *check* de paridade de erros e correcção de dados.



Figura 2.19 - Exemplo leitor RFID [7].

A função principal do *transponder* é guardar informações sobre o objecto em que está colocado [5]. É constituído por um circuito integrado ou um pequeno *chip* de silício, onde são armazenados os dados, uma antena, utilizada para a comunicação mas também para alimentação e o encapsulamento, que pode ser feito de vários materiais e ter várias formas. Os *transponders* podem ser classificados em três tecnologias, relativamente à sua alimentação, passivas, semi-passivas e activas. As etiquetas activas, usam bateria interna como fonte de alimentação, não precisam de energia do leitor para comunicar e um circuito de transmissão activo, o que garante um alcance muito maior (até 30 metros [8]). Por necessitarem de bateria para o seu funcionamento têm a desvantagem do tempo de vida ser limitado

(aproximadamente 10 anos), mesmo utilizando modos de poupança de energia. Dentro da classe de etiquetas activas, é comum designar-se a subclasse de etiquetas semi-activas. Estas ao contrário das activas, aproveitam a energia do leitor para comunicarem apresentando menor consumo. As etiquetas passivas, não possuem qualquer fonte de alimentação, utilizando a antena e o campo electromagnético criado pelo leitor para única e exclusiva fonte de energia. O seu alcance é por isso muito curto, e o seu tempo de vida é quase ilimitado. As etiquetas semi-passivas, tem um funcionamento muito semelhante aos das etiquetas passivas, estão dependentes do sinal do leitor para comunicar, no entanto são constituídas por uma bateria interna, esta tem como função fornecer energia ao circuito integrado e não para comunicar, permitindo ao circuito integrado uma maior capacidade de processamento, sendo por vezes equivalente a etiquetas activas.



Figura 2.20 - Exemplos de *transponder*.

A antena é um dispositivo que emite e recebe energia electromagnética, e existe em ambos os dois componentes anteriores, e tal como uma etiqueta pode ser interna ou externa. É o um componente imprescindível para a máxima eficiência e fiabilidade do sistema. A antena não funciona apenas como módulo de transmissão de dados, necessita também de transferir energia às etiquetas (passivas e semi-passivas). Quando campo electromagnético atravessa um condutor eléctrico, induz uma corrente eléctrica no condutor, sendo este efeito utilizado pela antena da etiqueta RFID de forma a fornecer energia eléctrica ao circuito integrado, para ser activa e realizar as operações de leitura/escrita e transmissão dos dados para a antena detectora (lei de *Faraday*). O princípio funcionamento é similar ao de um transformador, onde a antena transfere energia e os dados são trocados pelos dois elementos.

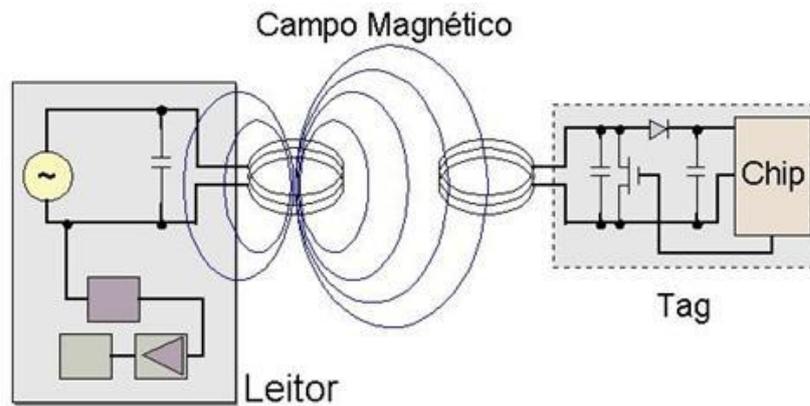


Figura 2.21 - Princípio de funcionamento do RFID.

Embora desta forma se descreva o funcionamento geral, existem diferentes modos de um leitor e etiqueta comunicarem. Por exemplo, os *transponders* de 1 único bit, utilizados em vigilância de artigos, a comunicação pode ser feita utilizando 5 técnicas diferentes de comunicação: divisor de frequência, microondas, radiofrequência, electromagnética e acústico-magnética [5]. Estas técnicas baseiam-se nos efeitos físicos e propriedades dos materiais, não existindo a troca de informação. As etiquetas de 1 bit são muito utilizadas nos hipermercados para prevenir roubos. O seu funcionamento consiste apenas em dois estados, activo ou desactivo. O estado activo significa que a etiqueta se encontra na zona de leitura do receptor, e no estado desactivo não há presença de etiqueta na zona de leitura. Todas as 5 técnicas seguem este princípio de identificação. Os *transponders* de maior capacidade de memória funcionam de forma diferente, pois é necessária a troca de dados. A troca de dados pode ser feita de duas formas, *half duplex* ou *full duplex*. No caso de *half duplex*, os dados transmitidos da etiqueta para o leitor é realizada após a transferência do leitor para a etiqueta. No caso *full duplex*, a troca de dados é realizada simultaneamente, isto implica que os dados transmitidos da etiqueta sejam feitos numa frequência diferente do leitor.

2.3.3 - Frequências de comunicação

A tecnologia RFID suporta várias frequências de comunicação entre os *transponders* e os leitores. Podem ser caracterizados pelas frequências que operam, como sendo, *Low*, *High*, *Ultra High* e *Super High Frequency*.

Tabela 2.1 - Classificação do Sistemas RFID relativamente à frequência de comunicação [9].

	LF (< 135KHz)	HF (13.56MHz)	UHF (863 até 915 MHz)	SHF (2.45 GHz)
Capacidade de dados	A partir de 64 bits para leitura até 2kbits para leitura e escrita.	Tipicamente etiquetas de leitura e escrita com 512 bits de memória (Max. 8kbits particionados).	Tipicamente etiquetas de leitura e escrita com 32 bits de memória (Max. 4kbits particionados em 128 bits).	A partir de 128 bits até 32 kbit particionados.
Transferência de dados	Baixa: menos de 1 kbit/s (~200bits/s).	Cerca de 25 kbit/s em geral (existe em 100 kbit/s).	Cerca de 28 kbit/s.	Geralmente <100 kbits/s, mas pode ir até 1 Mbit/s.
Distância de leitura	Tipicamente desde o contacto até 0,5 metros para etiquetas passivas, caso contrário, 2 metros aproximadamente.	Até 1 metro para etiquetas passivas.	Até 1 metro para etiquetas passivas.	Algumas dezenas de centímetros para etiquetas passivas e algumas dezenas de metros para as activas.
Modo de leitura	Leitura única e múltiplas leituras.	Leitura única e múltiplas leituras.	Leitura única e múltiplas leituras.	Leitura única e múltiplas leituras.
Ruído	Insensíveis às perturbações electromagnéticas industriais.	Ligeiramente sensíveis às perturbações electromagnéticas industriais.	Sensíveis às perturbações electromagnéticas. Podem ser perturbados por outros sistemas UHF próximos.	Altamente sensíveis a perturbações electromagnéticas reflectidas pelo metal e absorvidas pela água.
Aplicações	Processo de produção, identificação de veículos e contentores, controlo de acessos.	Monitorização estacionamento automóvel, livrarias, serviços de aluguer.	Monitorização estacionamento automóvel, logística.	Controlo de acessos, logística militar, portagens.

2.3.4 - Standardização de Protocolos RFID

O objectivo da standardização ou normalização, é definir plataformas em que uma dada indústria possa operar de forma segura e eficiente. A normalização é uma garantia de eficiência e portanto uma redução de custos. Na procura da standardização, diversas organizações estão envolvidas nos projectos de tecnologias RFID. As mais conhecidas na área dos sistemas RFID são a ISO (Internacional for Standardization) e a EPC Global.

No caso da EPC Global, foi a partir de projectos académicos realizados no Instituto de Tecnologia de Massachusetts (MIT), em parceria com algumas empresas, que se desenvolveu a tecnologia modelo para o rastreio e localização de produtos através de RF. O resultado destes projectos foi o EPC (Electronic Product Code). Para o funcionamento em conjunto com a tecnologia RFID, a EPC Global submeteu os seus protocolos e técnicas para aprovação à organização ISO, criando um conjunto de normas para estes sistemas.

A ISO é uma organização internacional de normalização, que desenvolve normas usadas mundialmente. Por isso, os protocolos são reconhecidos globalmente e utilizados como normas locais em vários países do mundo. O conjunto de normas referentes aos sistemas RFID, regulamentam todos os aspectos de funcionamento do sistema, desde a potência padrão para as antenas até como devem ser compostas as tramas de dados. Na Tabela 2.2, ilustra-se algumas normas publicadas pela ISO, aplicáveis aos sistemas RFID.

Tabela 2.2 - Normas publicadas pela ISO, aplicáveis aos sistemas RFID [14].

ISO Standard	Título	Estado
ISO 11784	RFID para animais - estrutura de código	Padrão publicado em 1996
ISO 11785	RFID para animais - concepção técnica	Padrão publicado em 1996
ISO/IEC 14443	Identificação de cartões - cartões com circuitos integrados sem contacto - cartões de proximidade	Padrão publicado em 2000
ISO/IEC 15693	Identificação de cartões - cartões com circuitos integrados sem contacto - cartões de vizinhança	Padrão publicado em 2000
ISO/IEC 18000-1	Parâmetros gerais para comunicação por interface por ar para frequências globalmente aceites	Padrão publicado em 2004
ISO/IEC 18000-2	Parâmetros gerais para comunicação por interface por ar abaixo de 135 kHz	Padrão publicado em 2004
ISO/IEC 18000-3	Parâmetros gerais para comunicação por interface por ar em 13,56 MHz	Padrão publicado em 2004
ISO/IEC 18000-4	Parâmetros gerais para comunicação por interface por ar em 2,45 GHz	Padrão em revisão final
ISO/IEC 18000-6	Parâmetros gerais para comunicação por interface por ar em 860 a 930 MHz	Padrão publicado em 2004
ISO/IEC 15961	Gestão de itens de RFID - Protocolo de dados: Interface de aplicação	Padrão publicado em 2004
ISO/IEC 15962	Gestão de itens de RFID - Protocolo de regras de codificação de dados e funções de memória lógica	Padrão publicado em 2004
ISO/IEC 15963	Gestão de Itens de RFID - Identificação única do RF Tag	Padrão em revisão final

2.4 - Outras Tecnologias de Identificação Automática

2.4.1 - Reconhecimento Óptico de Caracteres (OCR)

O reconhecimento óptico de caracteres (OCR, *optical character recognition*) converte imagens de texto como, por exemplo, documentos digitalizados, em caracteres de texto actuais. Também denominado reconhecimento de texto, o OCR permite editar e utilizar várias vezes o texto que normalmente está protegido dentro de imagens digitalizadas. O OCR funciona com o reconhecimento de padrões, para identificar caracteres de texto individuais numa página, incluindo marcas de pontuação, espaços e fins de linhas [10]. O OCR pode ser executado de três formas, automático, manual ou indexação.



Figura 2.22 - Exemplos de OCR.

2.4.2 - Visão Industrial

Os sistemas de visão industrial, são equipamentos de automação de captura e análise de imagens, que realizam medição e inspecção visual automática. As principais aplicações desta tecnologia são controlo de qualidade, medições, inspecção visual nas linhas de fabricação e embalagem. Esta tecnologia apenas recentemente foi classificada como tecnologia AIDC, devido à sua capacidade de leitura de códigos de barras lineares e bidimensionais. Oferece uma vantagem relativamente aos leitores de códigos de barras, pois permite obter o código de barras e o meio envolvente, dando origem a leituras com baixo contraste.



Figura 2.23 - Sistemas de visão industrial.

2.4.3 - Ecrãs tácteis

Os ecrãs tácteis são monitores que permitem a inserção de dados através do toque, dispensando assim a necessidade de outro periférico de entrada de dados, como o teclado. Podem ser classificados quanto à tecnologia utilizada na fabricação das telas, como sendo resistivos, capacitivos ou infravermelhos. Existem ainda ecrãs multi-toque, reconhecem múltiplos pontos de interacção em simultâneo. Dada a elevada robustez, são muito utilizados na indústria essencialmente para funções de controlo e supervisão.



Figura 2.24 - Exemplo de ecrãs tácteis.

2.4.4 - Cartões inteligentes (Smart cards)

Os cartões inteligentes, são cartões com características físicas muito semelhantes aos cartões de banda magnética. São constituídos por circuito integrado (IC) embutido, que consiste num microprocessador com uma interface SPI, de alimentação externa (leitor) e com um sistema operativo, por exemplo *JavaCard*. As soluções de cartões inteligentes oferecem soluções eficientes, flexíveis e móveis para verificação de identidade, encriptação de dados e validação de transacções para que as organizações possam diminuir os riscos de segurança, reduzir custos e oferecer aos utilizadores finais serviços convenientes e de valor acrescentado. A sua utilização tem vindo a aumentar ao longo dos anos, sendo os cartões telefónicos uma das maiores aplicações. São ainda utilizados no controlo de acessos, compra através cartão de crédito e verificação de assiduidade.

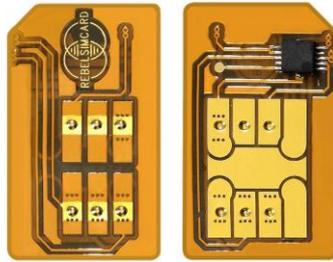


Figura 2.25 - Contactos IC.



Figura 2.26 - Exemplos de cartões inteligentes.



Figura 2.27 - Exemplo de leitores de cartões inteligentes.

2.4.5 - Tecnologias Biométricas

As tecnologias biométricas, são métodos de reconhecimento de pessoas baseado nas características físicas do corpo e dos traços pessoais. A tecnologia mais utilizada é o reconhecimento de impressões digitais, mas existem muitas outras, face, íris, retina, geometria da mão, ADN, veias, voz e caligrafia são alguns exemplos de métodos utilizados. Com a crescente aceitação e utilização da tecnologia biométrica particularmente no governo, serviços financeiros e sectores da saúde, o número de soluções é cada vez maior. A utilização desta tecnologia encontra-se presente nas mais diversas áreas, governo, indústria e empresas.



Figura 2.28 - Exemplo de tecnologias biométricas.

2.5 - Sumário das Tecnologias de Identificação e Recolha de Dados Automáticos

Na tabela 2.3, apresenta-se um breve sumário das tecnologias de identificação automáticas, abordadas no presente capítulo.

Tabela 2.3 - Sumário das diferentes tecnologias de identificação e recolha de dados automáticos [15].

Parâmetros do Sistema	Código de barras	OCR	Biométrica	Cartões Inteligentes	Sistemas RFID
Quantidade de dados (bytes)	1-100	1-100	-	16-64K	16-64K
Densidade de dados	Baixa	Baixa	Elevada	Muito elevada	Muito elevada
Legibilidade utilizando dispositivo de leitura	Bom	Bom	Custoso	Bom	Bom
Leitura realizada por pessoas	Limitado	Simples	Difícil	Impossível	Impossível
Influência de humidade/poeira	Muito elevada	Muito elevada	-	Possível, nos contactos	Não influência
Influência da direcção e posição	Baixa	Baixa	-	Unidireccional	Não influência
Degradação	Limitada	Limitada	-	Contactos	Não influência
Custo de aquisição	Muito baixo	Médio	Muito elevado	Baixo	Médio
Custos de operação (e.g. impressora)	Baixo	Baixo	Nenhum	Médio (contactos)	Nenhum
Modificação não autorizada	Leve	Leve	Impossível	Impossível	Impossível
Velocidade de leitura (incluindo manuseamento de dados)	Baixa ~4s	Baixa ~3s	Muito baixa > 5-10s	Muito baixa ~4s	Muito rápido ~0.5s
Máxima distância entre componente que contém os dados e o leitor	0-50 cm	Scanner <1 cm	Contacto directo*	Contacto directo	0-5 m

*Só se aplica para a identificação por impressão digital. No caso da íris ou da retina o contacto directo não é necessário.

Capítulo 3

Descrição do Problema

3.1 - Linha de Fabrico Flexível

No Departamento de Engenharia Electrotécnica de Computadores (DEEC), no laboratório de controlo e automação existe um simulador (Kit) de uma linha de fabrico flexível (Figura 3.1). É constituída por várias máquinas ferramenta, armazém, um robot de 3 eixos e vários sistemas de transporte baseado em tapetes rolantes (Figura 3.2). As peças que circulam na linha possuem etiquetas RFID embebidas, e existem 4 leitores para escrever e ler nas mesmas. Todos os sensores e actuadores estão ligados a um dispositivo de controlo (por exemplo autómato), sendo este responsável por toda a parte de controlo. A linha de fabrico flexível tem como principal objectivo, disponibilizar aos alunos um meio de aprendizagem de novas tecnologias de automação industrial. Os alunos têm assim a possibilidade de se familiarizarem com um sistema industrial muito próximo da realidade, e aplicarem os conhecimentos de automação leccionados nas unidades curriculares. Pretende-se integrar nesta solução, um conjunto de tecnologias de recolha e identificação automática de dados, dando a conhecer aos alunos mais tecnologias de automação, neste caso tecnologias AIDC.

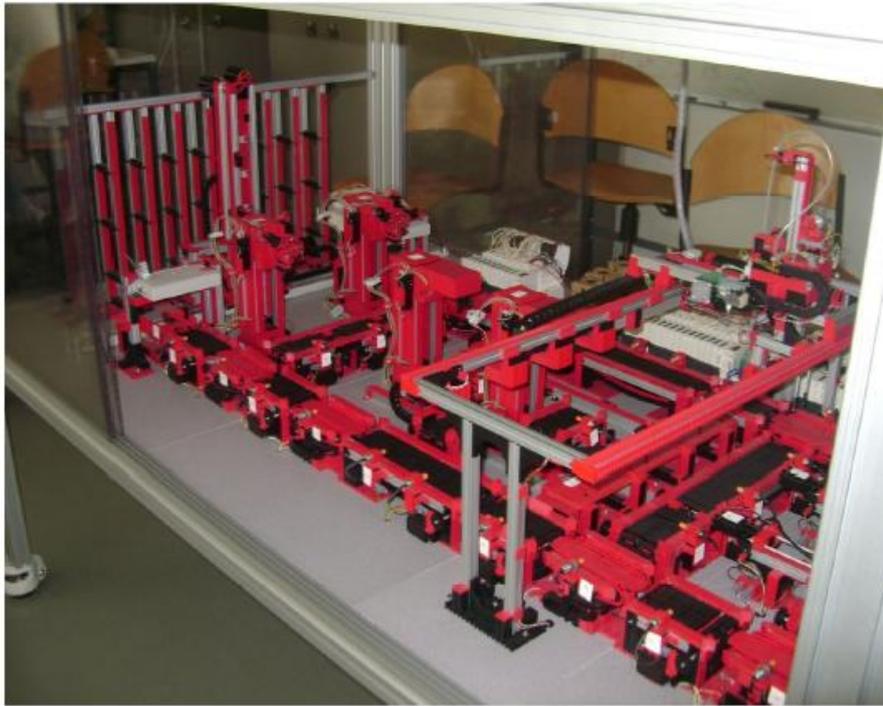


Figura 3.1 - Linha de fabrico flexível [5].



Figura 3.2 - Partes constituintes da linha de fabrico flexível [5].

3.2 - Análise de requisitos

Depois de apresentado o contexto geral do problema, serão apresentados de seguida os requisitos que se pretende atingir com a realização desta dissertação.

Pretende-se implementar uma plataforma que permita aos alunos utilizar e integrar um conjunto de tecnologias AIDC para utilização na linha de fabrico flexível, de forma simples e transparente. Os requisitos são os seguintes:

- Sistema RFID:
 - Ler dados de uma etiqueta com tamanho máximo de 8 bytes;
 - Escrever dados numa etiqueta, conhecida à priori a partir do seu UID, com tamanho máximo de 8 bytes. O UID identifica de forma unívoca a etiqueta, e tem tamanho 8 bytes;
 - Realizar a função de reset aos leitores RFID;
 - Obter o UID no máximo de 3 etiquetas.
- Leitor de Código de Barras:
 - Realizar a leitura de códigos de barras, da simbologia EAN-13.
- Impressora:
 - Imprimir códigos de barras simbologia EAN-13;
 - Imprimir texto com tamanho máximo de 1Kbyte;
 - Imprimir imagem em formato JPG, com tamanho máximo de 1Kbyte;
 - Imprimir comandos especiais, isto é, caracteres não imprimíveis, tais como *line feed*, *tab horizontal* e *carriage return*.
- Interface com utilizador:
 - Permitir aceder à plataforma, a partir de um cliente Modbus/TCP, para enviar comandos e consultar dados dos dispositivos de recolha e identificação automática de dados;
 - Desenvolver uma interface Web (cliente Modbus/TCP), que permita enviar comandos e consultar os dados dos dispositivos de recolha e identificação automática de dados.

Na Figura 3.3, apresenta-se a proposta da arquitectura do sistema que se pretende desenvolver.

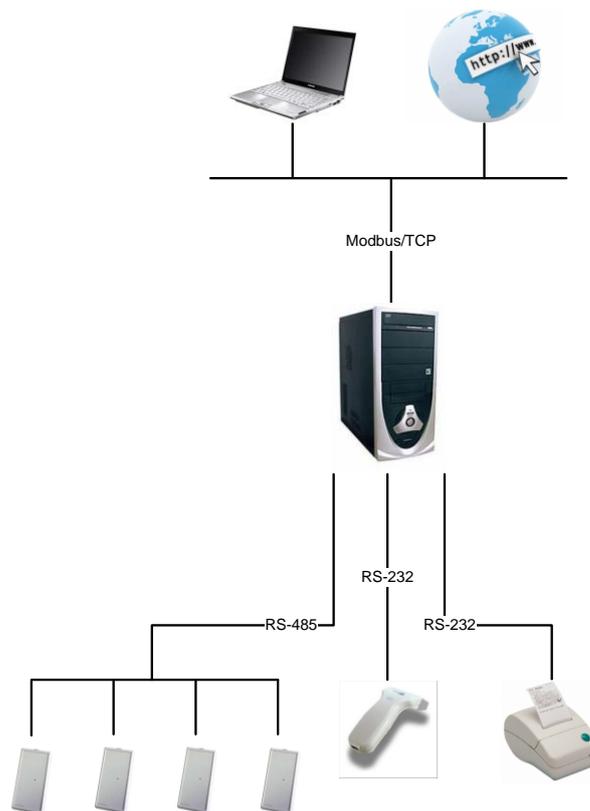


Figura 3.3 - Arquitectura do sistema com integração dos dispositivos AIDC.

Os clientes Modbus/TCP, servirão para disponibilizar duas interfaces, entre a plataforma e os dispositivos de recolha e identificação automática de dados. Desta forma, os alunos não necessitam de conhecer os protocolos de comunicação específicos dos diferentes dispositivos, terão apenas de enviar comandos, como por exemplo ler ou escrever numa etiqueta, para o sistema a partir da aplicação na plataforma.

Nas secções seguintes serão apresentadas as características e modos de funcionamento dos diferentes dispositivos que se pretende integrar, bem como o dispositivo onde serão integrados, isto é, a plataforma de integração. No caso dos leitores RFID, a sua escolha é inerente à linha de fabrico flexível, os restantes dispositivos já existiam e foi imposta a sua utilização no início da realização da dissertação.

3.3 - Leitor RFID

O leitor RFID utilizado na realização da dissertação é da marca FEIG modelo ID ISC.PR101-A (Figura 3.4). Tem uma indicação visual, um sinal luminoso constituído por um LED, que indica os 3 estados possíveis, verde significa funcionamento normal, vermelho quando detecta uma etiqueta e laranja quando o leitor inicializa. Na Tabela 3.1 apresenta-se as suas principais características.



Figura 3.4 - Leitor RFID modelo ID ISC.PR101-A.

Tabela 3.1 - Características do leitor RFID.

Frequência	13.56MHz
Tipo de etiqueta	Passiva
Antena	Interna
Alcance	Baixo (7 a 10 cm)
Interface de Comunicação	RS 232 ou RS-485
Ligação	9-pin D-sub fêmea
Alimentação	2 a 24V CC

O leitor RFID, apresenta 4 modos de funcionamento de transmissão de dados entre o leitor e o *host* (termo utilizado para designar dispositivo ao qual leitor se encontra ligado), sendo eles, *ISO15693 Host Comands*, *Scan Mode*, *Configuration Commands* e *Control Commands*. Os dois primeiros são utilizados para troca de dados entre etiqueta e o *host*, e os restantes utilizados para troca de dados do leitor. Os modos *Configuration Commands* e *Control Commands* consistem em comandos enviados pelo *host* para configuração e diagnóstico do leitor. Este responde com uma confirmação ou dados pedidos à ordem solicitada em caso de sucesso e com um erro em caso de insucesso.

Host (Terminal / PC / ...)		Reader	
parameter- / control command	→	parameter received and stored / control command processed	
		yes	no
	←	status / data	error status
	←		

Figura 3.5 - Troca de dados entre o *host* e o leitor RFID utilizando os modos *Configuration Commands* e *Control Commands* [11].

- *Configuration Commands*

Quanto à configuração (*Configuration Commands*), está organizada em blocos de 16 bytes, sendo 2 deles para CRC. Os blocos de configuração podem ser guardados, em dois locais distintos, na memória RAM ou na memória EEPROM. Na memória RAM, a configuração

enviada pelo *host* é actualizada imediatamente, e perde-se caso o leitor seja desligado. Na memória EEPROM, para que a configuração fique activa é necessário reinicializar o leitor, e esta não se perde após ser desligado. Os comandos disponíveis neste modo estão descritos na Tabela 3.2.

Tabela 3.2 - Funções disponíveis no modo de configuração.

Set Default Configuration	Coloca 1 ou todos os blocos de configuração com os valores de origem (RAM ou EEPROM)
Read Configuration	Lê um bloco de configuração (RAM ou EEPROM)
Write Configuration	Envia um bloco de configuração (RAM ou EEPROM)
Save Configuration	Guarda 1 ou todos os blocos de configuração na memória RAM ou EEPROM

- *Control Commands*

Quanto ao modo de controlo (*Control Commands*), permite realizar algumas funções tais como, determinar velocidade de transmissão, *reset* ao leitor, etc. Encontra-se na Tabela 3.3 as funções disponíveis neste modo de funcionamento.

Tabela 3.3 - Funções disponíveis no modo de controlo.

Baud Rate Detection	Determina a velocidade de comunicação (RS-232/RS-485)
Start Flash Loader	Inicia a actualização do firmware
CPU Reset	Reinicializa o CPU do leitor
Get Software Version	Determina a versão do software do leitor, o seu tipo e o tipo de etiquetas que são suportadas pelo software
Get Reader Info	Determina a versão do firmware do leitor, o seu tipo e o tipo de etiquetas que são suportadas pelo firmware
RF Reset	A antena é desligada por 15 milissegundos
RF ON/OFF	Liga ou desliga a antena
Set Output	Controla o LED

- *ISO15693 Host Comands*

O *ISO15693 Host Comands* consiste num conjunto de comandos que permite a troca de dados entre o *host* e as etiquetas através do leitor. A troca de dados pode ser feita de 3 formas diferentes:

- *Addressed Mode*: Antes de ler ou escrever numa etiqueta é necessário conhecer o seu UID. Para tal é enviado um comando *Inventory*, e se uma etiqueta estiver ao alcance do leitor retorna o seu UID. As funções de leitura e escrita seguintes para essa etiqueta têm de incluir o seu UID.

Host (Terminal / PC /)		Reader	
Inventory to get the UID	→	Transponder in antenna field ?	
		Yes	No
	←	status / number of Transponders / UID	
read data from Transponder with UID	→	Transponder with correct UID in antenna field ?	
		Yes	No
	←	status / Transponder read data	
write data to Transponder with UID	→	Transponder with correct UID in antenna field ?	
		Yes	No
	←	OK status	
	←	status = no Transponder in Reader field	

Figura 3.6 - Troca de dados entre host e o leitor RFID utilizando os modos *ISO15693 Host Comands: Addressed Mode* [11].

- *Non-Addressed Mode*: Não é necessário conhecer o UID da etiqueta, é útil quando há apenas uma etiqueta ao alcance do leitor.

Host (Terminal / PC /)		Reader	
read data	→	Transponder in antenna field ?	
		Yes	No
	←	status / Transponder read data	
write data	→	Transponder in antenna field ?	
		Yes	No
	←	OK status	
	←	status = no Transponder in Reader field	

Figura 3.7 - Troca de dados entre host e o leitor RFID utilizando os modos *ISO15693 Host Comands: Non-Addressed Mode* [11].

- *Selected Mode*: Tal como no *Non-Addressed Mode* é necessário enviar um comando *Inventory* para conhecer o UID da etiqueta. Posteriormente é enviado o comando *Selected* com esse UID. Assim a leitura e escrita para essa etiqueta que se realize a seguir não necessita de incluir o UID.

Host (Terminal / PC / ...)		Reader	
Inventory to get the UID	→	Transponder in antenna field ?	
		Yes	No
	←	status / number of Trans- ponders / UID	status = no Transponder
	←		
select Transponder with UID	→	Transponder with the correct UID in antenna field ?	
		Yes	No
	←	status / Transponder read data	status = no Transponder in Reader field
	←		
read data	→	selected Transponder in antenna field ?	
		Yes	No
	←	status / Transponder read data	status = no Transponder in Reader field
	←		
write data	→	selected Transponder in antenna field ?	
		Yes	No
	←	OK status	status = no Transponder in Reader field
	←		

Figura 3.8 - Troca de dados entre host e o leitor RFID utilizando os modos *ISO15693 Host Comands: Selected Mode* [11].

No modo de funcionamento *ISO15693 Host Comands*, os comandos oferecidos pelo leitor RFID dependem do tipo de etiqueta utilizada. Na Tabela 3.4, estão listadas as funções mais importantes, e que são suportadas pelo *transponder* utilizado no projecto.

Tabela 3.4 - Funções disponíveis no modo *ISO15693 Host Comands*.

Inventory	Retorna todos os UID das etiquetas ao alcance do leitor
Stay Quiet	Coloca a etiqueta em modo silencioso
Lock Multiple Blocks	Bloqueia um ou mais blocos de dados (contra escrita)
Read Multiple Blocks	Lê um ou mais blocos de dados
Write Multiple Blocks	Escreve num ou mais blocos de dados
Select	Selecciona a etiqueta (coloca em Select State)
Reset to Ready	Coloca a etiqueta em Ready State
Get System Information	Lê informação do sistema da etiqueta
Get Multiple Block Security Status	Lê o estado de segurança dos blocos de dados (se estão ou não protegidos contra escrita)

- *Scan Mode*

O *Scan Mode* consiste num modo automático de envio de dados da etiqueta para o *host* através do leitor RFID. Quando o *transponder* fica ao alcance do leitor, este envia de imediato a informação. Essa informação pode ser configurada quanto aos blocos de dados a enviar, UID da etiqueta, entre outras.

No que diz respeito ao *transponder* disponibilizado para a realização do projecto, não foi fornecida qualquer folha de características, mas a etiqueta é da *Texas Instruments* modelo *Tag-it HF-I Plus* (Figura 3.9).



Figura 3.9 - Peça da linha de fabrico flexível com *Tag-it HF-I Plus* embutida [5].

A *Tag-it HF-I Plus* é de baixa potência, *full duplex*. Tem a capacidade de 256 bytes de dados partilhados em blocos de 4 bytes, ou seja, 64 blocos de dados. Os dados são escritos e lidos usando a memória EPROM. Os blocos encontram-se separados e podem ser bloqueados para proteger os dados de alterações. Cada *transponder* tem um endereço único, designado UID, armazenado em 2 blocos programado de fábrica. Isto permite tratar de forma individual cada *transponder* e evitar colisões de uma multiplicidade de *transponders*.

3.4 - Leitor de Código de barras

O leitor de código de barras utilizado na realização da dissertação é o *scanner* manual CCD 1021Plus da marca Cipherlab (Figura 3.10). É alimentado a 5V CC, e possui interface RS-232, PS2 e *Wand Emulation*. Tem uma indicação visual, um sinal sonoro e luminoso, que indicam o seu respectivo estado e leituras bem sucedidas ou erros. Tem ainda um botão, utilizado para efectuar a captura de leituras.



Figura 3.10 - Scanner manual CCD 1021Plus.

O *scanner* manual CCD 1021Plus possui 8 modos diferentes de funcionamento:

- *Auto Off Mode*: Neste modo de funcionamento o *scanner* inicia a captura após o botão ser pressionado. A captura é contínua até que um código de barras tenha sido lido ou tenha sido atingido um determinado período de tempo (*scanner time-out*).
- *Continuous Mode*: O *scanner* está em permanente captura.
- *Auto Power Off Mode*: O *scanner* inicia a captura quando o botão é premido, tal como no *Auto Off Mode*, mas apenas termina quando expirar o *timeout*. Durante esse período de tempo poderá efectuar mais que uma leitura.
- *Alternate Mode*: A captura inicia quando o botão é pressionado, e termina quando for pressionado novamente.
- *Momentary Mode*: O *scanner* mantém a captura enquanto o botão estiver pressionado.
- *Repeat Mode*: Este modo de funcionamento é semelhante ao modo *Continuous Mode*, tendo o botão a função de retransmissão. Após uma leitura, se o botão for pressionado num intervalo de tempo de um segundo, os dados serão novamente retransmitidos tantas vezes quantas as vezes for pressionado o botão. Torna-se muito útil quando se pretende ler o mesmo código de barras várias vezes.
- *Laser Mode*: Este é o modo de funcionamento mais utilizado em *scanner* de *laser*. O *scanner* inicia a captura após o botão ser pressionado e termina quando for feita uma leitura, o botão for largado ou *timeout* expirar.
- *Test Mode*: O *scanner* está em permanente captura, mas ao contrário do *Continuous Mode*, descodifica repetidamente os códigos de barras, ou seja, poderá descodificar o mesmo código de barras várias vezes.

Relativamente aos códigos de barras, o *scanner* 1021Plus permite ler códigos de barras negativos. Normalmente os códigos de barras, as barras são representados por cor preta ou escura e os espaços por cor branca ou clara. Para os códigos de barras negativos é oposto, as barras são representadas com cores claras e os espaços com cores escuras.

O *scanner* suporta a maioria das simbologias, onde alguns dos seus parâmetros podem ser configuradas, as quais incluem:

- *Code 39 (Standart/Full ASCII)*
- *Italy Pharmacode*
- *French Pharmacode*
- *Industrial 25*
- *Interleave 25*
- *Matrix 25*
- *Codabar (NW-7)*
- UPCA
- UPCE
- EAN8
- EAN13
- *Code 93*
- *Code 128*
- EAN 128
- MSI
- *Plessey*

O leitor permite depois de descodificados os dados de uma leitura válida, configurar a trama que irá enviar para o *host*, desde inclusão de caracteres de início e fim de trama, tamanho da trama, identificação da simbologia, até criação de diferentes campos de dados dependendo dos requisitos da aplicação.

Todas as configurações do *scanner* são realizadas por leitura de códigos de barras disponibilizados no manual.

3.5 - Impressora

A impressora utilizada na realização da dissertação é da marca *Able Systems Limited* modelo Ap863-F (Figura 3.11). É alimentada a 12V CC a partir da bateria ou carregador externo, e possui uma interface de comunicação, RS-232. Tem uma interface visual com o utilizador, um sinal luminoso constituído por um LED, que indica os diferentes estados possíveis. A cor verde a piscar indica que a impressora está em funcionamento, enquanto a iluminação permanente significa que a bateria está a carregar. A cor verde confirma a operação normal. A cor laranja, significa que o papel está acabar ou impressora está no modo *Spooling Mode* e a cor vermelha significa que impressora está sem bateria. A ausência de luz

significa que a impressora está no modo hibernação, a bateria está descarregada ou não existe carregador externo ligado. É possível configurar outras combinações de cores do LED.

Tem ainda um botão, designado botão de alimentação de papel. Quando pressionado, com a impressora no modo *Sleep Mode*, o papel avança a uma velocidade de 50mm por segundo, dependendo da tensão de alimentação. Ao pressionar e soltar, tem ainda as seguintes funções, nos modos *Idle Mode* e *Spooling Mode*, o papel avança em passos de 1/16mm e quando no modo *Sleep Mode* a impressora entra no modo *Spooling Mode*. Com duplo *click*, isto significa pressionar e soltar de duas vezes consecutivas com um período semelhante ao de um rato de computador, se no modo *Idle Mode* imprime a mensagem de teste, se no modo *Spooling Mode* imprime os dados armazenados no *buffer*. A mensagem de teste, especifica versão de *firmware*, codificação, calibração dos dados, simbologias de código de barras suportados e conjunto completo de caracteres.



Figura 3.11 - Impressora *Able Systems Limited* modelo Ap863-F.

A impressora apresenta 3 modos de funcionamento:

- *Idle Mode*: A impressora encontra-se pronta para receber dados e imprimir de imediato. Ela responde ao botão de alimentação de papel, e pode imprimir a mensagem de teste. Geralmente a impressora muda automaticamente para o modo *Sleep Mode*, ao fim de um determinado período de tempo de se encontrar no modo *Idle Mode* sem realizar nenhuma impressão, para prolongar a duração da bateria.
- *Sleep Mode*: A impressora está efectivamente desligada, e acorda do modo de suspensão, após uma sequência de dados enviado pelo *host*, com uma certa quantidade de bits lógicos "0" a uma taxa de transmissão de 9600Bps, que corresponde normalmente a uma sequência de 50 caracteres zero, pressionar e soltar o botão de alimentação de papel ou ao conectar o carregador externo. Quando a impressora volta ao modo de suspensão, por comando do *host* ou após

um determinado período de inativação, caso contenha dados no *buffer*, estes são perdidos, apenas mantém as configurações.

- *Spooling Mode*: Os dados são recebidos e armazenados, mas não impressos. Este modo é utilizado, quando é necessário parar a impressão para a transferência de dados ou é necessário alterar o papel, os dados são armazenados no *buffer*. Em caso de falta de papel, a impressora muda para este modo de funcionamento de forma automática (por indicação de sensor óptico que indica ausência de papel), para evitar a perda de dados, e só abandona este modo de funcionamento clicando duas vezes no botão de alimentação de papel.

O *buffer* possui uma capacidade de 20Kbytes. A impressora permite imprimir texto, códigos de barras de diferentes simbologias, imagens e *tickets*.

3.6 - Plataforma de Integração

A plataforma de integração será constituída por um autômato programável. A sua utilização foi imposta no início da realização da dissertação. O autômato modular utilizado, é o *Modicon TSX Premium* da marca *Telemecanique* (Figura 3.12). É constituído pelos seguintes módulos:

- Fonte de alimentação: TSX PSY1620
- Processador: TSX 57202
- Carta de rede (*Ethernet*): TSX ETY 100 WS
- Carta de Comunicação: TSX SCY 21601
- PCMCIA Card:
 - TSX SCP 111 (RS-232): Módulo utilizado para integração do leitor código de barras no autômato.
 - TSX SCP 111 (RS-232): Módulo utilizado para integração da impressora no autômato.
 - TSX SCP 114 (RS-485): Módulo utilizado para integração dos leitores RFID no autômato.

Foi necessário incluir os módulos PCMCIA Card no autômato, para integrar no mesmo os diferentes dispositivos.



Figura 3.12 - Autômato *Modicon TSX Premium*.

É alimentado a 24V CC, e possui duas interfaces de comunicação, RS-232 e *Ethernet*. O autômato é um microcomputador com sistema operativo próprio, vocacionado para aplicações de controlo, programado pelo utilizador e funciona de forma cíclica. Existe um conjunto de linguagens que permite desenvolver os programas. As linguagens estão definidas numa norma, IEC 61131, a qual estabelece um padrão global para a programação de autômatos. A sua programação é realizada utilizando o *Software PL7*, desenvolvido pela *Telemecanique* e concebido para sistema operativo *Windows*, que respeita a norma referida. Para além de permitir realizar toda a programação, permite realizar todas as configurações necessárias nos diferentes módulos do autômato.

3.7 - Protocolo de comunicação Modbus

O protocolo Modbus/TCP é uma estrutura de mensagem desenvolvida em 1979 pela *Modicon*, e usada para estabelecer a comunicação entre dois dispositivos mestre-escravo/cliente-servidor. Os dispositivos Modbus comunicam utilizando a técnica mestre-escravo, a qual permite que somente um dispositivo, o mestre, inicie a comunicação. Os restantes dispositivos, os escravos, respondem de acordo com o pedido do mestre ou da tarefa em questão. O dispositivo escravo (válvula, sensor, dispositivo de medição), processa a informação e envia o dado para o mestre. É um protocolo de comunicação muito utilizado em dispositivos de automação, devido à sua simplicidade e versatilidade. Pode ser utilizado em diferentes meios físicos de comunicação, tal como, *Ethernet*, RS-232 ou RS-485. Os dados do Modbus podem ser de 4 tipos diferentes:

- *Discretes* - São variáveis de 1 bit e apenas podem ser lidas.
- *Coils* - São variáveis de 1 bit e podem ser lidas e escritas.
- *Registers* - São variáveis de 16 bits e apenas podem ser lidas.
- *Holding registers* - São variáveis de 16 bits de leitura e escrita.

Para aceder a estas variáveis, o protocolo Modbus define endereços de 0 a 65535. O Modbus/TCP consiste numa rede de múltiplos dispositivos ligados por uma rede TCP/IP num modelo mestre-escravo. Um mestre envia um pedido para um cliente e este responde com a informação pedida. Esta troca é feita por meio de tramas Modbus/TCP numa conexão TCP previamente estabelecida entre mestre e escravo. O estabelecimento e gestão de conexões são feitos pelo protocolo TCP/IP e ocorrem independentemente do protocolo Modbus. Alguns dispositivos servidores em Modbus, suportam múltiplas conexões de clientes simultaneamente. Da mesma forma, os clientes podem estabelecer múltiplas conexões, com servidores distintos. Adicionalmente o protocolo Modbus/TCP permite a um cliente enviar múltiplos pedidos Modbus ao mesmo servidor sob a mesma conexão, mesmo antes da resposta aos pedidos anteriores. Um dispositivo pode ser cliente e servidor simultaneamente.

Construção da trama TCP/IP-*ethernet*:

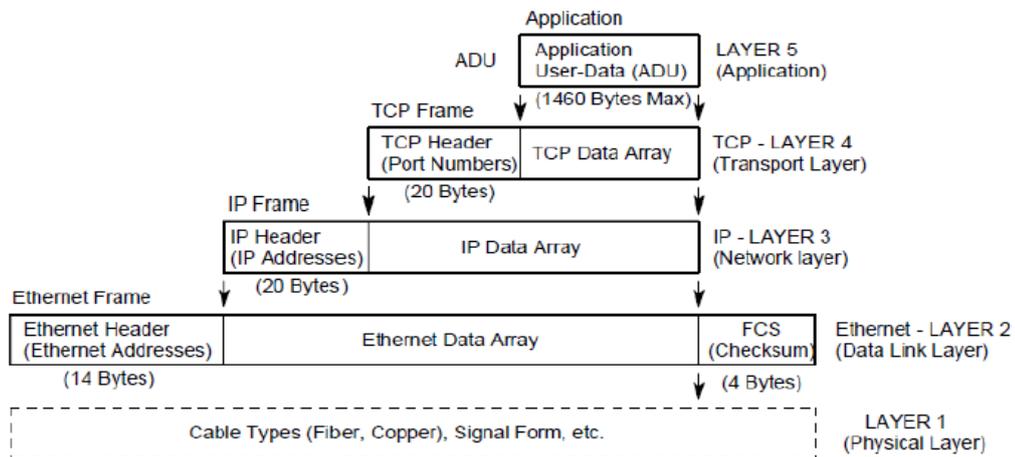


Figura 3.13 - Construção da trama TCP/IP-*ethernet*.

No caso concreto descrito, a *Application Layer* corresponderá ao Modbus. Desta forma uma trama Modbus terá o seguinte aspecto:

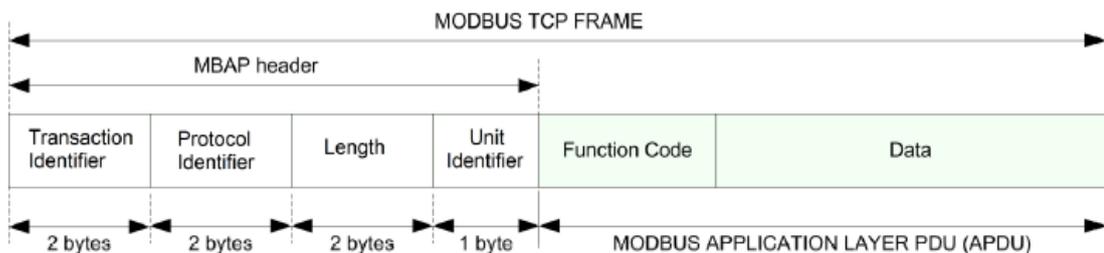


Figura 3.14 - Trama Modbus/TCP.

A trama consiste num cabeçalho MBAP (*Modbus Application Protocol*), mais a camada de aplicação PDU (*Protocol Data Units*). O cabeçalho MBAP é formado por:

- *Transaction Identifier*: Uma vez que um cliente pode efectuar múltiplas transacções seguidas sob uma só conexão TCP, não há garantias que as respostas cheguem na mesma ordem dos pedidos. Torna-se então necessária inclusão de um identificador para cada transacção, de forma associar o pedido com a resposta. Este campo é inicializado pelo cliente, faz um pedido e repetido pelo servidor de resposta.
- *Protocol Identifier*: Usado para identificar o protocolo.
- *Length*: Tamanho em bytes do *Unit ID* mais os da aplicação PDU. Serve para identificar os limites da trama.
- *Unit Identifier*: Identificador do dispositivo destinatário da mensagem.

Notar que as tramas Modbus/TCP não possuem um campo de detecção de erros (ao contrário do que se passa com o Modbus), pois tal não é necessário, uma vez que a *stack* TCP/IP já inclui mecanismos de detecção de erros. Na figura seguinte ilustra-se as diferenças entre a trama Modbus e a trama Modbus/TCP.

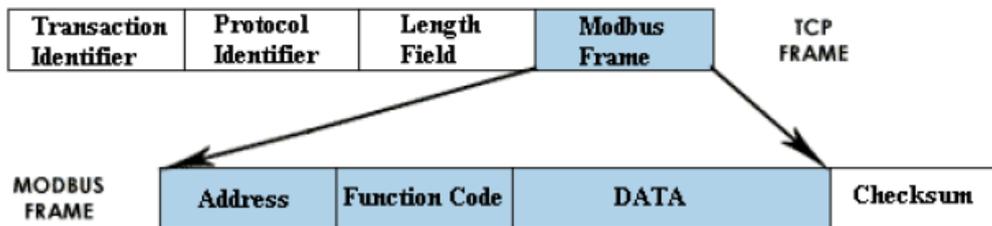


Figura 3.15 - Diferenças entre trama Modbus e Modbus/TCP.

Na tabela seguinte, ilustra-se as funções de leitura/escrita de saídas/entradas ou variáveis internas usadas na trama de comunicação do protocolo Modbus/TCP.

Tabela 3.5 - Funções do protocolo de comunicação Modbus/TCP.

				Function Code		
				Code	Subcode	(Hex)
Data Access	Bit Access	Input Bits	Read input bits	02		02
			Read internal bits	01		01
		Internal Bits	Write internal bits	05		05
			Write internal bit chain	15		0F
	Word Access (Register)	Input Word	Read register	04		
			Read word table	03		
		Internal Word	Write register	06		
			Write word table	16		10
			Sync. read/write of tables	23		17
			Flag register	22		16
	Data Set	Read data set	20	6	14	
		Write data set	21	6	15	
Interface		Read Device Identification	43	14	28	

3.8 - Servidor Modbus no Autómatu Programável

Como referido anteriormente, o sistema terá dois clientes Modbus/TCP e um servidor Modbus/TCP. O autómatu é o servidor Modbus/TCP do ponto de vista exterior, mas cliente Modbus/TCP do ponto de vista dos dispositivos de identificação e recolha de dados automáticos, excepto para o leitor de código de barras. Este, é quem toma a iniciativa de comunicar com o autómatu.

O servidor Modbus/TCP está implementado automaticamente no autómatu. O protocolo Modbus, fornece um meio pelo qual um dispositivo pode ler e escrever nos registos de outro dispositivo remoto. A organização dos registos no servidor, para que o cliente possa ler e escrever, é definida pelo protocolo Modbus. Os pedidos por parte do cliente Modbus/TCP, correspondem a ler posições de memória. Existe uma relação directa entre os registos Modbus e as posições de memória do autómatu, tal como ilustrado na Figura 3.16, onde %M são as *Memory Bits* e os %MW são as *Memory Word*. Desta forma, é possível uma abstracção dos dispositivos por parte do autómatu.

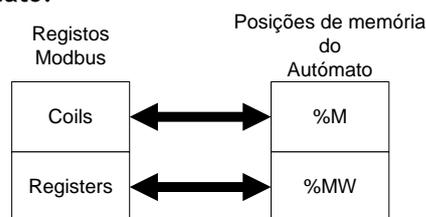


Figura 3.16 - Relação entre registos Modbus e posições de memória do autómatu.

Capítulo 4

Proposta da Aplicação

4.1 - Introdução

No presente capítulo, será discutida a estrutura da aplicação adoptada para alcançar a solução do problema apresentado no capítulo anterior.

No servidor Modbus/TCP, será reservada uma zona de registos, onde se encontra organizada a informação de todos os dispositivos de recolha e identificação automática de dados. Esta solução foi adoptada, pois o servidor Modbus/TCP já se encontra implementado no autómato, de forma automática. Para além disso, o próprio protocolo define a organização da memória no autómato, para que os clientes Modbus/TCP possam ler e escrever os dados de forma transparente. Desta forma os clientes Modbus/TCP acedem ao servidor Modbus/TCP, podendo executar as funções Modbus, tais como ler e escrever, de forma transparente.

Nas tabelas seguintes ilustra-se, a organização dos registos Modbus bem como a descrição dos mesmos para os diferentes dispositivos. Note-se que os registos estão classificados como de leitura (L) e leitura e escrita (L/E). A organização dos registos foi pensada, tendo em conta alguns factores tais como, a organização dos dados por tipo de dispositivo, e em cada dispositivo, sempre que necessário, nos primeiros registos encontram-se os respectivos comandos e erros, e de seguida os dados de leitura ou escrita.

- **Registos dos Leitores RFID**

Tabela 4.1 - Organização dos registos Modbus para os leitores RFID.

	Endereço (s)	Registo	Tipo	Descrição
	300	CMD/STATUS_RFID	L/E	<p>Leitor 1: bits 0,1 e 2.</p> <ul style="list-style-type: none"> • Bit 0 e 1: função a realizar pelo leitor RFID. • '00': Nenhuma Função • '10': Inventory • '01': Read Multiple Blocks • '11': Write Multiple Blocks <p>• Bit 2: Quando a '1' erro, não existe etiqueta ao alcance do leitor ou erro na resposta ao pedido.</p> <p>Leitor 2: bits 3,4 e 5. Leitor 3: bits 6,7 e 8. Leitor 4: bits 9, 10 e 11.</p>
	301	CMD_WRITE/STATUS_RFID	L/E	<p>Leitor 1: bits 0, 1, 2 e 3.</p> <ul style="list-style-type: none"> • Bits 0, 1 e 2: indica a etiqueta onde se pretende escrever os dados. • Bit 3: Quando a '1' erro, etiqueta seleccionada incorrecta ou não existe etiqueta seleccionada. <p>Leitor 2: bits 4, 5, 6 e 7. Leitor 3: bits 8, 9, 10 e 11. Leitor 4: bits 12, 13, 14 e 15.</p>
Leitor 1	302-305	ID_TAG_1	L	UID tag 1
	306-309	ID_TAG_2	L	UID tag 2
	310-313	ID_TAG_3	L	UID tag 3
	314-317	BUFFER_L1	L/E	Dados a ler ou escrever da tag 1
Leitor 2	318-321	ID_TAG_1	L	UID tag 1
	322-325	ID_TAG_2	L	UID tag 2
	326-329	ID_TAG_3	L	UID tag 3
	330-333	BUFFER_L2	L/E	Dados a ler ou escrever da tag 2
Leitor 3	334-337	ID_TAG_1	L	UID tag 1
	338-341	ID_TAG_2	L	UID tag 2
	342-345	ID_TAG_3	L	UID tag 3
	346-349	BUFFER_L3	L/E	Dados a ler ou escrever da tag 3
Leitor 4	350-353	ID_TAG_1	L	UID tag 1
	354-357	ID_TAG_2	L	UID tag 2
	358-361	ID_TAG_3	L	UID tag 3
	362-365	BUFFER_L4	L/E	Dados a ler ou escrever da tag 4

As funções que se podem executar nos leitores RFID são, de leitura (*Read Multiple Blocks*), escrita (*Write Multiple Blocks*) e inventário (*Inventory*). Para os clientes Modbus/TCP solicitarem a execução das funções, tem de escrever no registo CMD/STATUS_RFID. Neste registo são indicadas as funções a realizar por cada um dos leitores RFID, e alguns erros associados. Os bits 0 e 1, são utilizados para indicar a função a realizar pelo leitor 1, os bits 3 e 4 pelo leitor 2, os bits 6 e 7 pelo leitor 3 e os bits 9 e 10 pelo leitor 4. São suficientes os 2 bits por cada leitor RFID, para solicitar a execução das diferentes funções disponíveis. No bit 2, indica-se o erro do leitor 1, no bit 5 do leitor 2, no bit 8 do leitor 3 e no bit 11 do leitor 4. O erro indica, que não existe etiqueta ao alcance do leitor ou resposta ao pedido incorrecta. Se a função a realizar pelos leitores for escrita, é necessário indicar no registo CMD_WRITE/STATUS_RFID, a etiqueta onde se pretende escrever bem como introduzir no respectivo registo os dados a ser escritos. A selecção da etiqueta no leitor 1, corresponde a colocar o bit 0, 1 ou 2 a '1', o bit 4,5 ou 6 no leitor 2, o bit 8,9 ou 10 no leitor 3 e o bit 12,13 ou 14 no leitor 4. Apenas é possível, a escrita numa etiqueta cada vez que se executa a função de escrita. O bit 3 do leitor 1, o bit 7 do leitor 2, o bit 11 do leitor 3 e o bit 15 do leitor 4, indicam ocorrência de erro de escrita, isto é, quando a etiqueta seleccionada está incorrecta ou quando não existe nenhuma etiqueta seleccionada para proceder à escrita. Os registos BUFFER_L1, BUFFER_L2, BUFFER_L3 e BUFFER_L4 contem os dados dos diferentes leitores. Se a função a realizar for leitura, contem os dados lidos, se a função a realizar for escrita deve conter os dados a escrever. O tamanho dos dados é de 8 bytes, isto é 2 blocos de dados, podendo ser modificado.

Os registos ID_TAG_1, ID_TAG_2 e ID_TAG_3, contem os UID das etiquetas, onde cada UID tem o tamanho de 8 bytes.

- **Registos do Código de Barras**

Tabela 4.2 - Organização dos registos Modbus para o leitor de código de barras.

Endereço (s)	Registo	Tipo	Descrição
366-372	BUFFER_SCANNER	L	Código de barras capturado pelo scanner

No leitor de código de barras não é necessário qualquer registo quer para os comandos, quer para os erros, apenas é necessário o registo para os dados. O registo BUFFER_SCANNER, contem os 13 dígitos que correspondem à leitura realizada pelo *scanner*. Cada dígito ocupa 1 byte.

- **Registos da Impressora**

Tabela 4.3 - Organização dos registos Modbus para impressora.

Endereço (s)	Registo	Tipo	Descrição
373	CMD/STATUS_IMP	L/E	Bits 0 e 1: Função a realizar pela impressora. '00': Nenhuma função '10': Impressão de código de barras '01': Impressão de texto '11': Impressão de imagem Bit 2: Carriage Return. Bit 3: Line Feed. Bit 4: Tab Horizontal. Bit 5: Quando a '1' erro, caracter não imprimível.
374-874	BUFFER_IMP	E	Código de barras, texto ou bitmap a ser impresso

No registo CMD/STATUS_IMP é indicada a função a realizar pela impressora e os erros associados. Os bits 0 e 1 são utilizados para indicar a função a realizar pela impressora. Os bits 2, 3 e 4 são utilizados para indicar a impressão de comandos não imprimíveis. Quando se pretende realizar alguma das funções dos comandos não imprimíveis, é necessário indicar nos bits 0 e 1 a função de impressão de texto. Desta forma, supondo que se pretendia realizar a função de *line feed*, o valor a escrever no registo CMD/STATUS_IMP deve ser 0x0A. O bit 5, indica a ocorrência de erro.

O registo BUFFER_IMP, é o registo utilizado para escrever os dados, caso seja necessário, a utilizar por cada uma das diferentes funções disponíveis na impressão. No caso da impressão de código de barras, no registo BUFFER_IMP é escrito os 12 dígitos do código de barras. Note-se que é necessário escrever no registo CMD/STATUS_IMP o valor 0x01, para indicar a função de impressão de código de barras. Se no registo BUFFER_IMP, não forem introduzidos apenas dígitos, o valor lido no registo CMD/STATUS_IMP será, 0x21, indicando a ocorrência de erro.

- **Funcionamento geral da Aplicação**

Os registos como ilustrado, contém os dados, os comandos e os possíveis erros dos diferentes dispositivos. Estes serão actualizados constantemente de forma cíclica. A aplicação a implementar na plataforma, será constituída pelas diferentes funções que permitem actualizar os registos dos dispositivos de identificação e recolha de dados automáticos. As funções implementadas na aplicação, permitem uma reorganização parcial de forma automática dos registos, quando se pretende adicionar ou remover algum dispositivo do autómato ou alterar o número de blocos dados, sendo apenas necessário a configuração de alguns parâmetros, tal como, o número de leitores RFID ou o número de blocos de dados a ler e escrever. Os diferentes parâmetros são configurados no início das funções da aplicação, semelhante a um cabeçalho. Na tabela 4.4, ilustra-se a reorganização dos registos para todos

os dispositivos, quando se altera o número de blocos de dados de 2 para 1 nos leitores RFID. De forma semelhante, acontece quando se adiciona ou remove algum dispositivo no automático. Esta reorganização automática não é possível na sua totalidade, por limitação do automático. O número máximo de dispositivos adicionar, é determinado pelos bits disponíveis nos registos de comando dos diferentes dispositivos, isto é, no registo que indica a função a realizar por cada um deles. Assim no sistema RFID, apenas é possível adicionar mais um leitor, pois no registo `CMD/STATUS_RFID`, estão apenas disponíveis os 4 bits mais significativos, sendo necessários 3 destes bits para o leitor RFID adicionar. No caso do registo `CMD_WRITE/STATUS_RFID`, seria criado um novo registo, `CMD_WRITE/STATUS_RFID_2`, que apenas ficava ocupado nos 4 bits menos significativos, que correspondem ao leitor adicionado.

Tabela 4.4 - Reorganização automática dos registos Modbus para os todos os dispositivos.

	Endereço (s)	Registo
	300	CMD/STATUS
	301	CMD_WRITE/STATUS
Leitor 1	302-303	ID_TAG_1
	304-305	ID_TAG_2
	306-307	ID_TAG_3
	308-309	BUFFER_L1
Leitor 2	310-311	ID_TAG_1
	312-313	ID_TAG_2
	314-315	ID_TAG_3
	316-317	BUFFER_L2
Leitor 3	318-319	ID_TAG_1
	320-321	ID_TAG_2
	322-323	ID_TAG_3
	324-325	BUFFER_L3
Leitor 4	326-327	ID_TAG_1
	328-329	ID_TAG_2
	330-331	ID_TAG_3
	332-333	BUFFER_L4
	334-340	BUFFER
	341	CMD/STATUS
	342-842	BUFFER

As funções dos diferentes dispositivos implementadas na aplicação, são tratadas de forma independente. Na Figura 4.1, ilustra-se o fluxograma que descreve de forma geral o funcionamento da aplicação. Quando a aplicação inicia o seu funcionamento, executa a função de *reset* aos leitores RFID, de seguida executa a função pretendida em cada um dos

leitores RFID, de seguida executa a função pretendida na impressora e verifica se o *scanner* realizou a aquisição de uma nova leitura de um código de barras. Irá repetir ciclicamente, a execução das diferentes funções nos diferentes dispositivos.



Figura 4.1 - Fluxograma que descreve o princípio de funcionamento da aplicação.

4.2 - Integração Leitores RFID

4.2.1 - Introdução

O sistema RFID, será composto por 4 leitores. Para a integração dos leitores RFID no autómato, o meio físico de comunicação a utilizar, será um barramento RS-485 (*half-duplex*). O standard RS-485, foi desenvolvido para atender à necessidade de comunicação multiponto, e permite a conexão até 32 dispositivos. Dado que pretendemos integrar 4 dispositivos, torna-se necessário endereçar os leitores RFID. O endereço tem o tamanho de 1 byte. O leitor 1 foi endereçado com o endereço 0X01, o leitor 2 com endereço 0X02, o leitor 3 com o endereço 0X03 e por fim o leitor 4 com o endereço 0X04.

O leitor RFID possui 2 protocolos de comunicação de troca de dados com o autómato, o *Standart Protocol* e *Advanced Protocol*.

- *Standart Protocol*: Suporta dados até 255 bytes. Se este for o protocolo de comunicação utilizado entre o autómato e o leitor, a trama de resposta dependerá dos dados a enviar. Se a trama não ultrapassar os 255 bytes utilizará o *Standart*

Protocol, caso contrário, utilizará o *Advanced Protocol*. A estrutura da trama de envio e de resposta está definida na Figura 4.2 e na Tabela 4.5.

- *Advanced Protocol*: Suporta dados superiores a 255 bytes. Neste caso se este for o protocolo de comunicação utilizado entre o autómato e o leitor, a resposta será sempre em *Advanced Protocol*. A estrutura da trama de envio e de resposta está definida na Figura 4.3 e na Tabela 4.5.

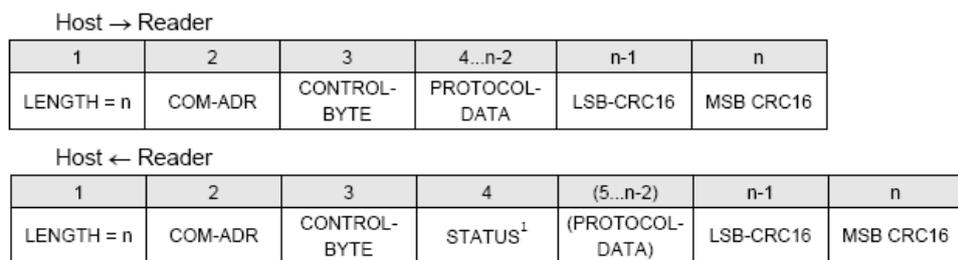


Figura 4.2 - Estrutura das tramas do protocolo *Standart Protocol* [11].

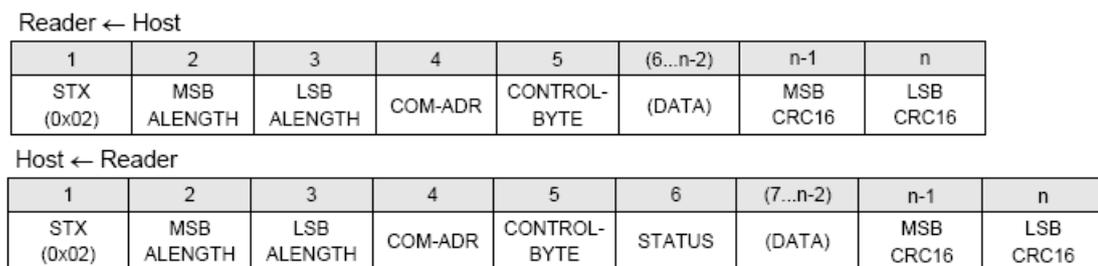


Figura 4.3 - Estrutura das tramas do protocolo *Advanced Protocol* [11].

Tabela 4.5 - Descrição da trama do protocolo de comunicação do leitor RFID.

Byte (s)	Descrição
STX	Indica uma trama com mais de 255 bytes (<i>Advanced Protocol</i>)
ALENGTH	Número de bytes da trama incluindo STX, ALENGHT e CRC16
LENGTH	Número de bytes da trama incluindo LENGHT e CRC16
COM-ADR	Endereço do leitor caso esteja ligado a um barramento (RS-485)
CONTROL-BYTE	Define o comando a executar
STATUS	Inclui uma mensagem de estado do leitor
DATA	Campo opcional com tamanho variável (depende do comando)
CRC16	Polinómio $X^{16} + X^{12} + X^5 + 1$ Valor inicial 0xFFFF

Dos protocolos comunicação apresentados e suportados pelo leitor RFID, utilizou-se o *Standard Protocol*, pois a troca de dados que se espera para a aplicação não ultrapassará os 255 bytes, e porque o Modbus não permite enviar ou receber tantos dados de uma só vez, apenas permite no máximo 250 bytes. A partir deste ponto, sempre que se referir a estrutura das tramas, subentende-se que o protocolo utilizado é o *Standard Protocol*.

No capítulo anterior apresentaram-se os 4 diferentes modos de funcionamento de transmissão de dados entre o leitor e o *host*. Quanto ao protocolo, 3 destes modos apenas diferem no *Control-Byte* (comando) e no campo de dados (*data*). O *Scan Mode* não utiliza nenhum dos protocolos descritos, pois a comunicação é feita apenas num sentido, do leitor para o autómato, e os dados são enviados numa trama simples, que consiste em caracteres de início, separação e fim, com a possibilidade de ser configurado.

4.2.2 - Funções Implementadas na Aplicação

Para satisfazer os requisitos que correspondentes aos leitores RFID, foram implementadas na aplicação as seguintes funções:

- *CPU Reset*: Permite realizar reset aos leitores RFID.
- *Inventory*: Permite obter o UID das etiquetas.
- *Read Multiple Blocks (non-addressed mode)*: Permite ler dados até no máximo 8 bytes, isto é, 2 blocos de dados.
- *Write Multiple Blocks (addressed mode)*: Permite escrever dados até no máximo 8 bytes, numa etiqueta conhecida à priori, a partir do seu UID.

Na Figura 4.4 ilustra-se o funcionamento do sistema RFID. O sistema, tem apenas de verificar qual a função a executar em cada um dos leitores, e executar a função pretendida em cada um deles. Começa por verificar no leitor com o endereço 0x01, 0x02, 0x03 e por fim 0x04, tal como ilustrado na Figura 4.4.

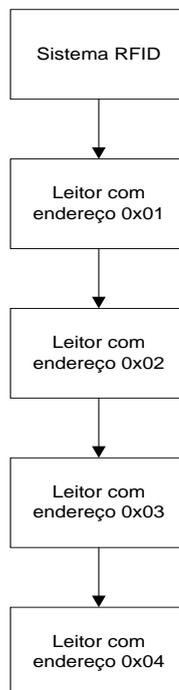


Figura 4.4 - Fluxograma que descreve funcionamento do sistema RFID.

Na Figura 4.5, ilustra-se o fluxograma que descreve o funcionamento do leitor RFID com o endereço 0x01. Se no registo CMD/STATUS_RFID se escrever 0x00 o leitor não executa nenhuma função, se escrever 0x01 o leitor executará a função *Inventory*, se escrever 0x02 executará a função *Read Multiple Blocks* e por fim se escrever 0x03 executará a função *Write Multiple Blocks*. Para os restantes leitores RFID o princípio de funcionamento é o mesmo, apenas se altera o valor a escrever no registo CMD/STATUS_RFID, que deve ser escrito nos bits correspondentes ao leitor RFID, tal como ilustrado na secção anterior. Assim, por exemplo, se pretender executar a função *Read Multiple Blocks* no leitor 3 deve-se escrever no registo CMD/STATUS_RFID 0x80, e se pretender executar a função *Inventory* no leitor 4 escrever 0x200. De forma semelhante, acontece para o registo CMD_WRITE/STATUS_RFID, quando se pretende seleccionar a etiqueta para proceder à escrita.

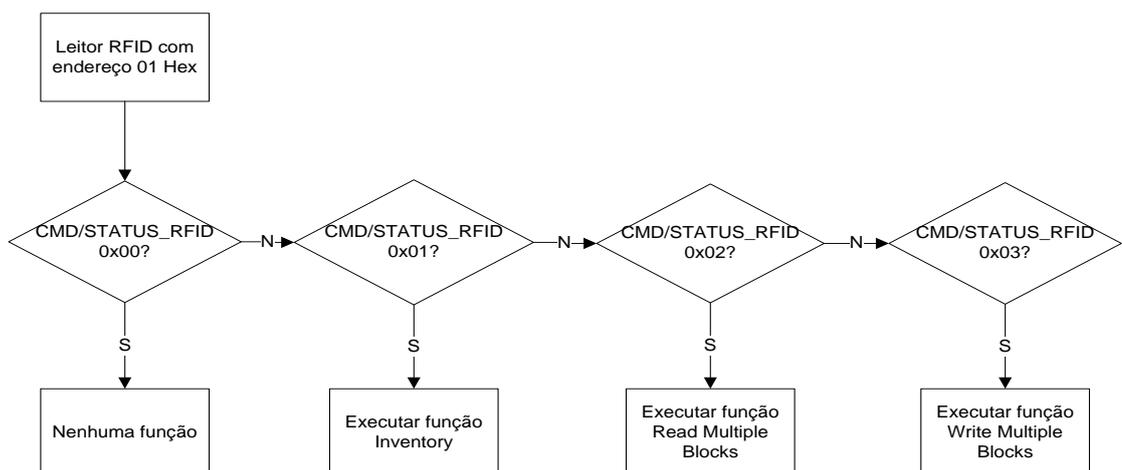


Figura 4.5 - Fluxograma que descreve funcionamento do leitor RFID com endereço 0x01.

Nas seguintes figuras, ilustra-se os fluxogramas que descrevem o funcionamento das funções implementadas na aplicação.

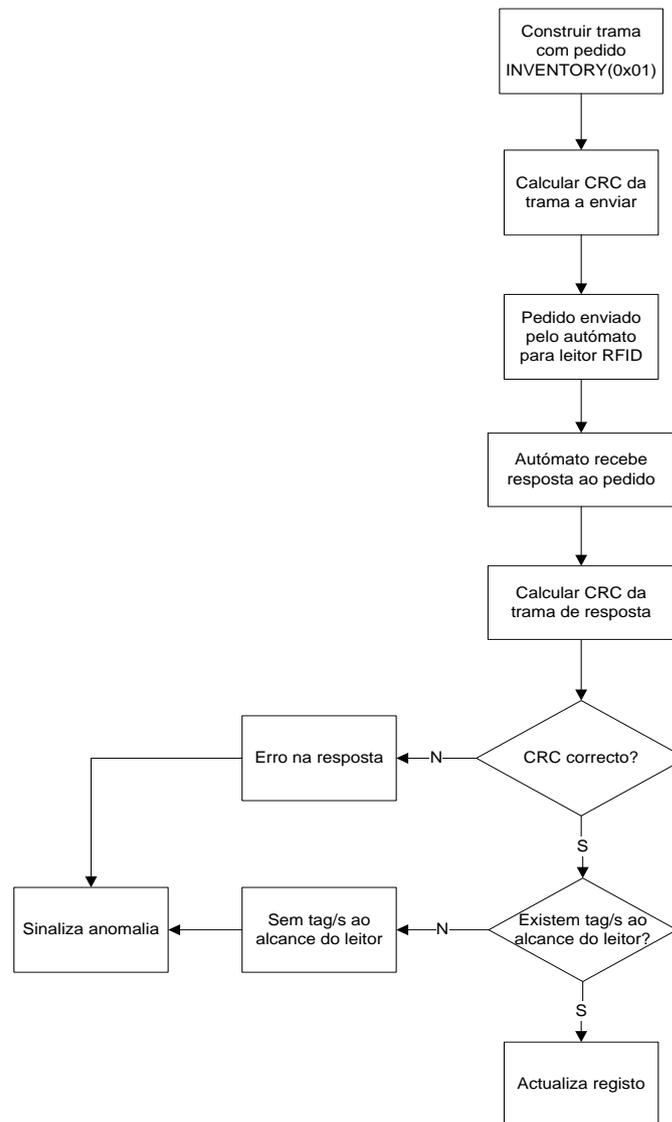


Figura 4.6 - Fluxograma da função *Inventory*.

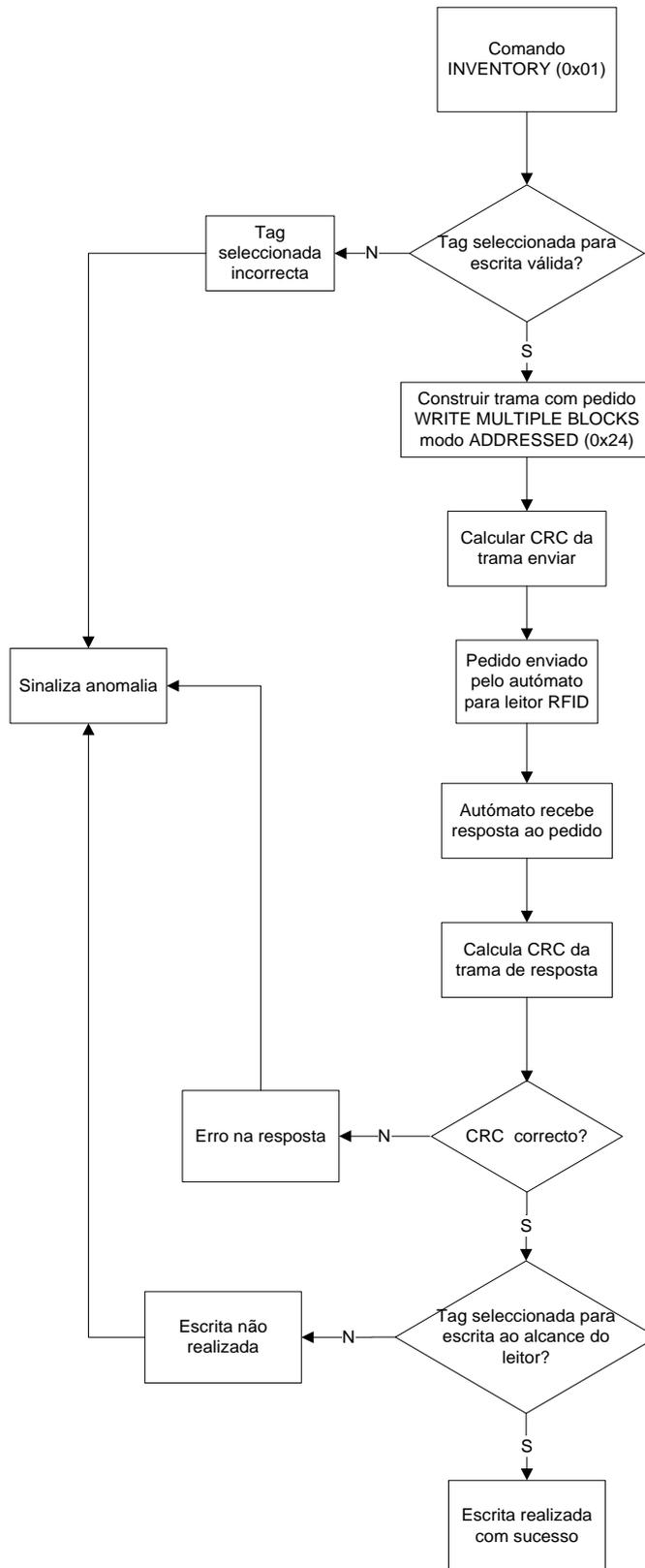


Figura 4.7 - Fluxograma da função *Write Multiple Blocks*.

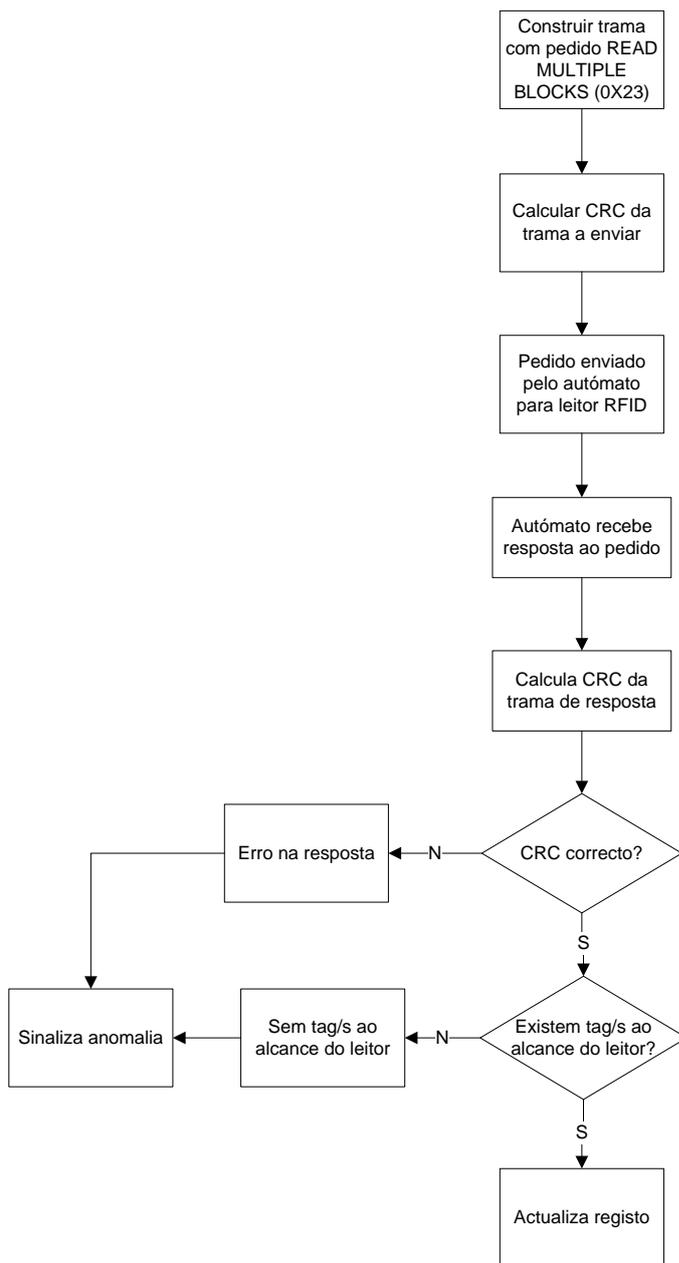


Figura 4.8 - Fluxograma da função *Read Multiple Blocks*.

Nas funções *Inventory* (Figura 4.6) e *Read Multiple Blocks* (Figura 4.8), o seu funcionamento é semelhante. Construi-se a trama com pedido, o autómato envia o pedido para o leitor RFID respectivo, e aguarda uma resposta. Quando recebida a resposta, verifica se a mesma está correcta. Se estiver errada retorna um erro, caso contrário analisa a resposta recebida. Se a resposta não contiver os dados pretendidos, isto é, os dados de resposta ao pedido efectuado, retorna um erro, caso contrário actualizado o respectivo registo no autómato.

A função *Inventory* (0x01), permite obter o UID da etiqueta. Pode obter-se o UID de várias etiquetas apenas com execução de um comando, desde que estas se encontrem ao

alcance do leitor. O UID da etiqueta tem tamanho de 8 bytes, e permite a sua identificação de forma unívoca. A estrutura da trama está definida na Figura 4.9 e na Tabela 4.6.

Assim, se a função a realizar pelo leitor 2 for *Inventory*, a trama enviada pelo autómato para o leitor será '07 02 B0 01 00 B8 AA' em hexadecimal. A resposta ao pedido deve conter pelo menos o UID de uma etiqueta, '11 02 B0 00 01 03 00 E0 07 80 AC DD E7 29 5A 48 64' em hexadecimal, onde os 8 bytes 'E0 07 80 AC DD E7 29 5A' corresponde ao UID da etiqueta.

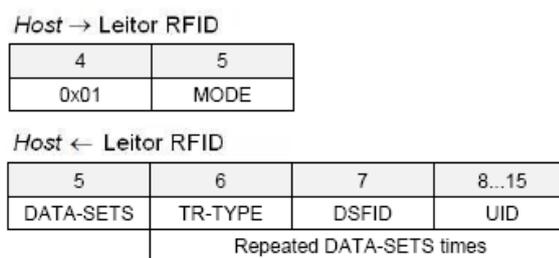


Figura 4.9 - Trama de envio e resposta do comando *Inventory* [11].

Tabela 4.6 - Descrição da trama *Inventory*.

Byte (s)	Descrição
MODE	Indica se comando <i>Inventory</i> executado para novo UID ou para UID já existente
DATA-SETS	Número de etiquetas
TR-TYPE	Indica a tecnologia da etiqueta no que diz respeito à frequência de comunicação
DSFID	Indica família de dados (ISO15693)
UID	Identificação da etiqueta

Na função *Write Multiple Blocks* (Figura 4.7), o princípio de funcionamento é diferente, dado que se pretende o modo *Addressed Mode*. Começa por executar a função *Inventory* para obter os UID das etiquetas ao alcance do leitor. De seguida, verifica se etiqueta seleccionada para efectuar a escrita é válida, caso contrário retorna erro. Se etiqueta seleccionada é válida, constrói a trama com pedido de escrita na etiqueta seleccionada. O autómato envia pedido para o respectivo leitor e aguarda uma resposta. Quando recebida a resposta, se estiver errada retorna erro, caso contrário verifica se dados foram escritos com sucesso. Se os dados não foram escritos com sucesso, retorna um erro.

Os comandos *Read Multiple Blocks* (0x23) e *Write Multiple Blocks* (0x24), permitem ler e escrever blocos de dados nas etiquetas. O tamanho dos blocos de dados varia com o tipo de etiqueta, sendo neste caso de 4 bytes. A estrutura das tramas para estes comandos está definida na Figura 4.10, Figura 4.11 e na Tabela 4.7.

Se a função a realizar pelo leitor 3 for *Read Multiple Blocks*, a trama enviada pelo autómato para o leitor 3 será, '09 03 B0 23 00 00 02 C3 E9' em hexadecimal. A resposta será '12 03 B0 00 02 04 00 32 30 32 30 00 32 30 32 30 8C 8B' em hexadecimal. No leitor 4, se a função a realizar for *Write Multiple Blocks*, na etiqueta 'E0 07 80 AC DD E8 2F 6F' em

hexadecimal, com a escrita dos dados '30 30 30 30 30 30 30' em hexadecimal, a trama enviada pelo autómato para o leitor será, '1A 04 B0 24 01 E0 07 80 AC DD E8 2F 6F 00 02 04 30 30 30 30 30 30 AE 99' em hexadecimal. Se a escrita for realizada com sucesso a resposta será, '11 04 B0 00 01 03 00 E0 07 80 AC DD E8 2F 6F 1C 12' em hexadecimal.

Host → Leitor RFID

3	4	5	(6...13)	6 / (14)	7 / (15)
[0xB0]	0x23	MODE	UID	DB-ADR	DB-N

Host ← Leitor RFID

3	4	5	6	7	8...n
[0xB0]	STATUS	DB-N	DB-SIZE	SEC-STATUS	DB
Repeated DB-N times					

Figura 4.10 - Trama de envio e resposta do comando *Read Multiple Blocks* [11].

Host → Leitor RFID

3	4	5	(6...13)	6 / (14)	7 / (15)	8 / (16)	9...n / (17...n)
[0xB0]	0x24	MODE	UID	DB-ADR	DB-N	DB-SIZE	DB

Host ← Leitor RFID

3	4
[0xB0]	STATUS

Repeated DB-N times

Figura 4.11 - Trama de envio e resposta do comando *Write Multiple Blocks* [11].

Tabela 4.7 - Descrição das tramas *Read Multiple Blocks* e *Write Multiple Blocks*.

Byte (s)	Descrição
MODE	Indica modo de comunicação com a etiqueta (Non-addressed, Addressed, Selected)
UID	Identificação da etiqueta
DB-ADR	Endereço do primeiro bloco de dados a ser lido ou escrito
DB-N	Número de blocos de dados a ser lido ou escrito
DB-SIZE	Número de bytes de cada bloco de dados
SEC STATUS	Indica se bloco de dados está protegido ou não
DB	Dados a ser lidos ou escritos de 1 bloco

A função para efectuar reset aos leitores RFID é simples, é apenas necessário construir a trama e enviar a mesma, do autómato para os diferentes leitores RFID. Na Figura 4.12, está definida a estrutura da trama para este comando. Assim, a trama enviada pelo autómato para o leitor 1 será, '05 01 63 CB 48' em hexadecimal.

Host → Reader

1	2	3	4,5
5	COM-ADR	[0x63]	CRC16

Figura 4.12 - Trama de envio do comando CPU Reset [11].

A construção das diferentes tramas inclui o cálculo do respectivo CRC. Na aplicação foi implementada uma função, que calcula o CRC. A mesma função é utilizada para verificar se a resposta recebida no autómato dos diferentes leitores RFID, está correcta. O facto da resposta recebida no autómato estar correcta, não implica uma resposta correcta ao pedido realizado pelo mesmo.

Os diferentes exemplos das tramas, foram apresentadas para um leitor RFID, mas para os restantes, na construção das tramas, apenas se altera o endereço do leitor RFID e o respectivo CRC.

4.2.3 - Meio Físico de Comunicação

Como referido anteriormente, o meio físico de comunicação a utilizar para a integração os 4 leitores RFID, será um barramento RS-485 (*half-duplex*), tal como ilustrado na Figura 4.13. Na Figura 4.14, ilustra-se as ligações físicas necessárias realizar, nos leitores RFID e no módulo de comunicação TSX SCP 114 do autómato. Note-se que o pino 7 e 9 dos leitores RFID, correspondem à sua alimentação.

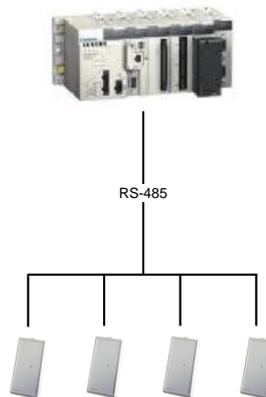


Figura 4.13 - Integração dos leitores RFID no autómato.

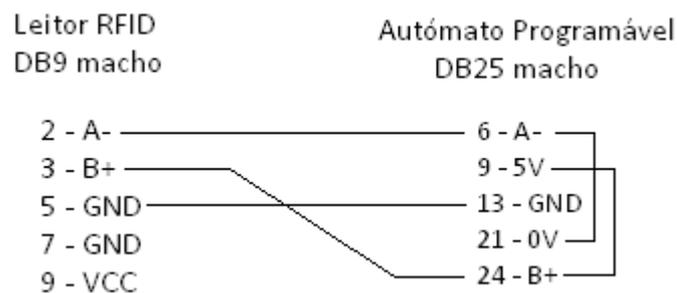


Figura 4.14 - Esquema de ligações do leitor RFID no autómato.

4.3 - Integração do Leitor de Código de Barras

4.3.1 - Introdução

A integração do leitor de código de barras é de instalação fácil, sendo apenas necessário configura-lo. Desta forma, o primeiro e único passo consiste na sua configuração. O leitor de código de barras será configurado para a leitura de códigos de barras com simbologia EAN-13, e não se pretende a inclusão de diferentes campos de dados, apenas os dados referentes à leitura realizada. O leitor suporta várias simbologias, podendo ser alterada a qualquer altura, sendo apenas necessário a sua reconfiguração. O modo de funcionamento escolhido foi *Auto Off Mode*, onde o *scanner* inicia a aquisição após botão ser pressionado.

Como mencionado no capítulo anterior, todas as configurações do *scanner* são realizadas por leitura de códigos de barras disponibilizados no manual.

4.3.2 - Função implementada na Aplicação

A única função a realizar pelo leitor de código de barras, é a aquisição de novas leituras. Assim que o botão do *scanner* é premido inicia-se a aquisição, e caso um código de barras seja lido é enviado para o autómato. O autómato recebe a leitura realizada pelo *scanner*, e verifica se a mesma é válida, a partir do seu comprimento, dado que o comprimento esperado da trama é de 13 dígitos. Se a leitura for válida o respectivo registo será actualizado, caso contrário o registo mantém a última leitura válida. A trama enviada pelo *scanner* para o autómato, corresponde apenas aos 13 dígitos da leitura em hexadecimal. Desta forma, se o código de barras lido pelo *scanner* for '5601312045755', a trama enviada pelo *scanner* para o autómato será '35 36 30 31 33 31 32 30 34 35 37 35 35' em hexadecimal.

A função implementada na aplicação, tem apenas de verificar se o *scanner* realizou a aquisição de uma nova leitura.

4.3.3 - Meio Físico de Comunicação

O meio físico de comunicação a utilizar para integrar o *scanner*, será a comunicação série RS-232 (Figura 4.15). Na Figura 4.16, ilustra-se as ligações físicas necessárias realizar no leitor de código de barras e no módulo de comunicação TSX SCP 111 do autómato.

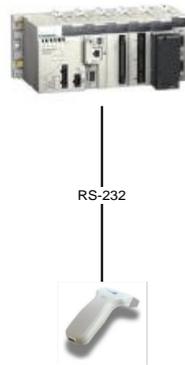


Figura 4.15 - Integração do leitor código de barras no autômato.

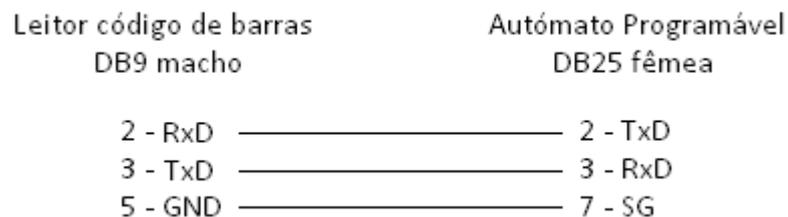


Figura 4.16 - Esquema de ligação do *scanner* no autômato.

4.4 - Integração da Impressora

4.4.1 - Introdução

Após se ter estudado o princípio de funcionamento da impressora e se ter realizado alguns testes básicos, procedeu-se análise dos comandos que permitem construir as tramas com os respectivos pedidos. Todos os comandos disponíveis bem como a sua respectiva função encontram-se descritos no manual da impressora [12]. De todos os comandos estudados, apenas será apresentado os comandos mais importantes para a aplicação.

- *Line Feed*: '0A' em hexadecimal
- *Carriage Return*: '0D' em hexadecimal
- Acordar a impressora do modo de suspensão: '1D 52 00' em hexadecimal
- Impressão código de barras 012345678901 simbologia EAN-13: '1D 6B 02 (simbologia EAN-13=02), 30 31 32 33 34 35 36 37 38 39 30 31 (código de barras em hexadecimal), 00' em hexadecimal
- Texto automático do código de barras: '1D 48 01' em hexadecimal (acima do código de barras), '1D 48 02' em hexadecimal (abaixo do código de barras)
- Definir altura do código de barras: '1D 68 01' em hexadecimal (n*0,125mm, neste caso n=1)

- Definir largura do código de barras: '1D 72 01' em hexadecimal (n*0,125mm, neste caso n=1)
- Imagem bit map: '1B 2A 00 ... (imagem em hexadecimal)' em hexadecimal

Admitindo que se pretende imprimir o código de barras 567009234518, simbologia EAN-13, com 0,125mm*0,125mm, com texto automático abaixo do código e com a impressora no modo de funcionamento *Sleep Mode*, a trama que o autómato deverá enviar para a impressora será a seguinte:

'1D 52 02 1D 68 01 1D 72 01 1D 48 02 1D 6B 02 35 36 37 30 30 39 32 33 34 35 31 38 00'
em hexadecimal.



Figura 4.17 - Código de barras impresso.

4.4.2 - Funções Implementadas na Aplicação

Para satisfazer os requisitos correspondentes à impressora, foram implementadas na aplicação as seguintes funções:

- Impressão de Códigos de barras simbologia EAN-13, com texto automático abaixo do código
- Impressão de texto
- Impressão imagens
- Impressão de comandos não imprimíveis, *line feed*, *tab horizontal* e *carriage return*.

Na figura 4.18, ilustra-se o fluxograma que descreve o funcionamento da impressora. Se no registo CMD/STATUS_IMP se escrever 0x01, a impressora executará a função de impressão de código de barras, se escrever 0x02 executará a função de impressão de texto e se escrever 0x03 executará a função de impressão de imagem. Para executar a função dos comandos especiais, terá de se escrever no registo CMD/STATUS_IMP um dos 3 valores possíveis. Se escrever 0x06 executará o comando *carriage return*, se escrever 0x0A executará *tab horizontal* e se escrever 0x12 executará *line feed*.

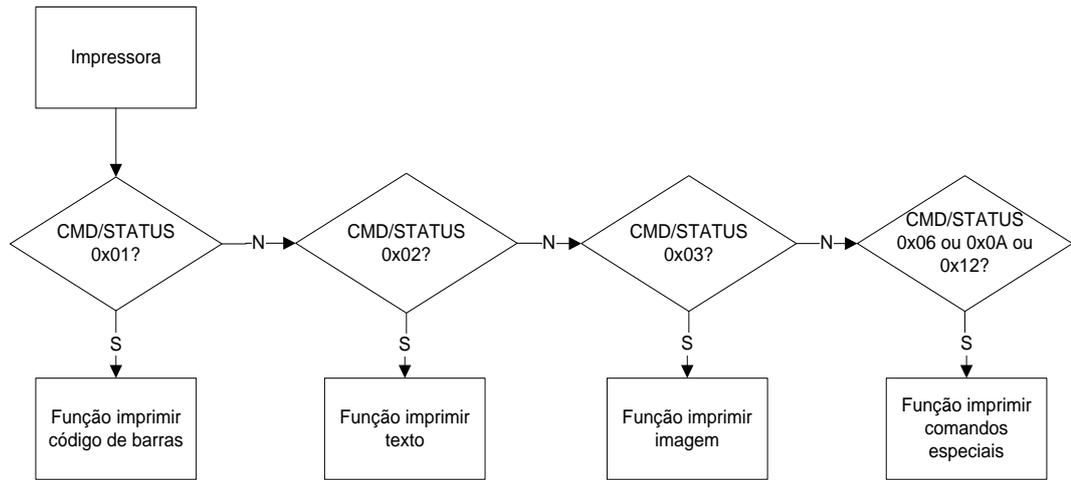


Figura 4.18 - Fluxograma que descreve aplicação da impressora.

Nas seguintes figuras, ilustra-se os fluxogramas que descrevem o funcionamento das funções implementadas na aplicação.

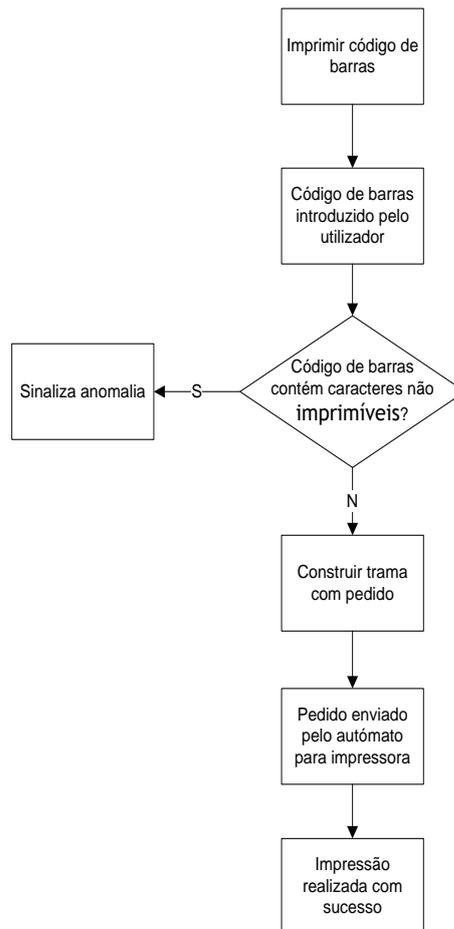


Figura 4.19 - Fluxograma da função imprimir código de barras.

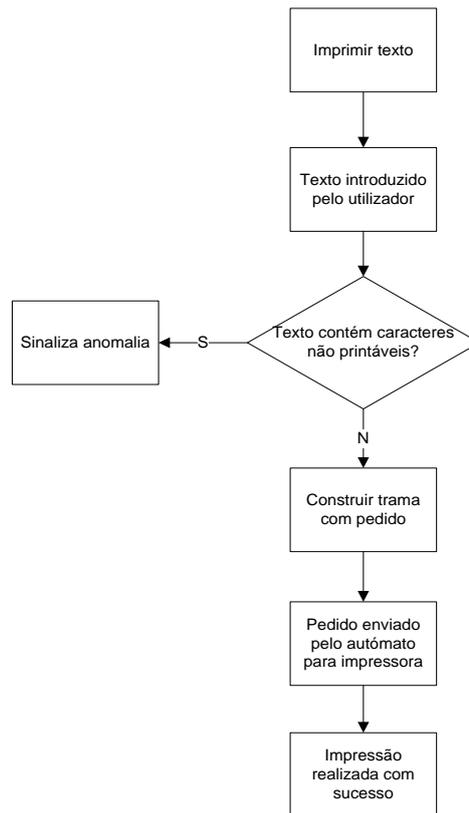


Figura 4.20 - Fluxograma da função imprimir texto.

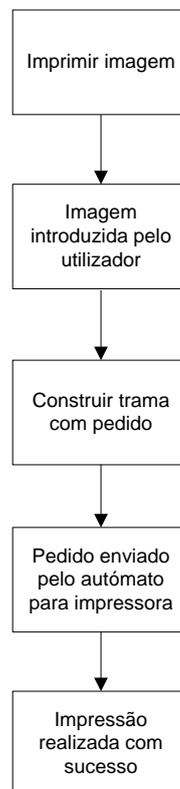


Figura 4.21 - Fluxograma da função imprimir imagem.



Figura 4.22 - Fluxograma da função imprimir comandos especiais.

As funções para impressão de código de barras (Figura 4.19) e texto (Figura 4.20), tem um princípio de funcionamento muito semelhante. Começa por verificar se os dados introduzidos pelo utilizador contém caracteres não imprimíveis, e se existirem retorna um erro. Se não existir caracteres não imprimíveis, é construída a trama com o respectivo pedido. Depois de construída a trama, o autómato envia a mesma para a impressora, e a impressão é realizada com sucesso. Se pretender imprimir o código de barras 560116308071, com 0,125mm*0,125mm e com texto automático abaixo do código, a trama enviada pelo autómato para a impressora será, '1D 68 01 1D 72 01 1D 48 02 1D 6B 02 35 36 30 31 31 36 33 30 38 30 37 31 00' em hexadecimal. Se pretender imprimir, por exemplo o texto 'Faculdade de Engenharia', a trama enviada pelo autómato para a impressora será, '1B 4A 70 23 00 46 61 63 75 6C 64 61 64 65 20 45 6E 67 65 6E 68 61 72 69 61' em hexadecimal.

As funções para impressão de imagem (Figura 4.21) e comandos especiais ou não imprimíveis (Figura 4.22), tem também um princípio de funcionamento semelhante. Iniciam com a construção da trama, com os dados introduzidos pelo utilizador. Depois de construída a trama, o autómato envia a mesma para a impressora, e a impressão é realizada com sucesso. Para impressão de imagem, a trama enviada pelo autómato para a impressora será '1B 2A 00 dados da imagem em hexadecimal' em hexadecimal. No caso dos comandos não imprimíveis a trama enviada pelo autómato para a impressora será, '0D' em hexadecimal para *carriage return*, '0B' em hexadecimal para *tab vertical* e '0A' em hexadecimal para *line feed*. Note-se que as diferentes tramas são compostas por comandos.

4.4.1 - Meio Físico de Comunicação

O meio físico de comunicação a utilizar para integrar a impressora, será a comunicação série RS-232 (Figura 4.23). Na Figura 4.24, ilustra-se as ligações físicas necessárias realizar, na impressora e no módulo de comunicação TSX SCP 112 do autómato.

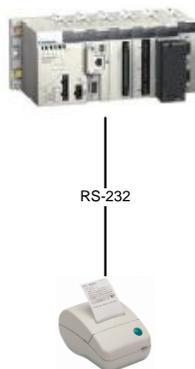


Figura 4.23 - Integração da impressora no autómato.

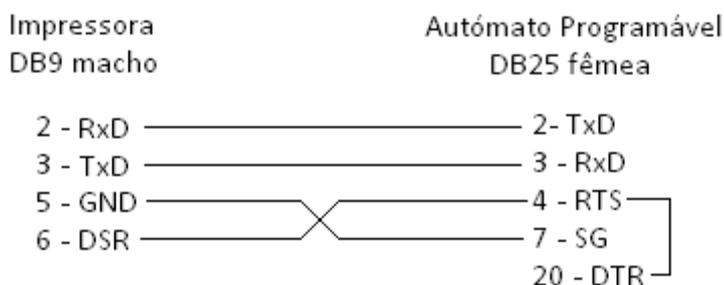


Figura 4.24 - Esquema de ligações da impressora no autómato.

4.5 - Funções do Autómato Programável

Para a troca de dados entre o autómato e os diferentes dispositivos de recolha e identificação automática de dados, utilizaram-se algumas funções já disponibilizadas pelo autómato para este efeito. Das funções disponibilizadas utilizaram-se, a *print_char*, a *out_in_char* e *input_char*. É necessário especificar alguns parâmetros, que servem de parâmetros de entrada da função, tais como, dados a enviar, comprimento dos dados a enviar, comprimento dos dados a receber e o designado *report*, que permite “monitorizar” a comunicação entre o autómato e o dispositivo.

A função *print_char*, permite apenas enviar os dados do autómato para um dispositivo. Esta função é utilizada quando se pretende realizar a função de reset aos leitores RFID, pois apenas é necessário enviar a trama com o respectivo pedido para os leitores RFID.

A função *input_char*, permite enviar os dados do dispositivo para o autómato. É utilizada na função implementada na aplicação, que permite enviar a leitura capturada pelo *scanner* para o autómato.

A função *out_in_char*, além de permite enviar os dados do autómato para um dispositivo, permite receber os dados enviados do dispositivo para o autómato. Esta função é utilizada na maioria das funções desenvolvidas na aplicação, dado que pretendemos enviar um pedido do autómato para o dispositivo, mas também receber no autómato a resposta do dispositivo ao pedido.

4.6 - Interface Web

4.6.1 - Objectivo

O objectivo da interface Web, é permitir interagir com o sistema, através de um browser. A interface permitirá enviar comandos e consultar os dados dos dispositivos de identificação e recolha de dados automáticos.

4.6.2 - Desenvolvimento da Aplicação

A aplicação foi desenvolvida utilizando as linguagens de programação Web, HTML, PHP e Javascript. O HTML é uma linguagem utilizada para produzir páginas Web estáticas, enquanto o PHP é utilizado para gerar conteúdo dinâmico. A linguagem Javascript complementa as capacidades do HTML, permite a interacção do utilizador com a página Web e a validação dos formulários do lado do cliente, melhorando muito as aplicações Web.

Dado que o protocolo de comunicação a utilizar será Modbus/TCP, são necessárias as funções de leitura (03) e escrita (16) deste protocolo, para aplicação Web poder escrever e ler dados do servidor Modbus/TCP. Para tal, utilizou-se a biblioteca PHPModbus [13], que já disponibiliza as funções necessárias mas em UDP, sendo apenas necessário modificar para TCP.

Uma vez que interface está disponível a qualquer utilizador, sendo apenas necessário um browser, criaram-se 2 tipos de utilizadores. Se o utilizador se encontrar autenticado, poderá consultar os dados, bem como enviar comandos para o sistema, caso contrário apenas poderá consultar os dados. Desta forma garante-se, que apenas os utilizadores habilitados podem enviar comandos para o sistema. A aplicação disponibiliza ainda a validação de todos os formulários do lado do cliente, isto é, não permitirá submeter comandos ao servidor, caso estes estejam incorrectos ou incompletos.

A aplicação é constituída pelos seguintes menus:

- **Consultar dados RFID:** Permite consultar os dados dos leitores RFID, tais como, UID das etiquetas, dados a ler ou escrever, função que o leitor se encontra a executar e possíveis erros. Antes de visualizar os dados, é necessário seleccionar o leitor ou os leitores, que se pretende consultar os dados. A página Web onde se apresenta os dados, encontra-se constantemente em actualização, a uma taxa de 5 segundos, para actualizar os dados dos leitores RFID com frequência.



Figura 4.25 - Interfaces gráficas dos menus Consultar dados RFID.

- **Comandos RFID:** Permite enviar comandos para realizar as funções nos leitores RFID. No caso da função *Write Multiple Blocks*, deve ser inserido os dados a escrever, bem como seleccionar a etiqueta onde se pretende escrever. Para as restantes funções, é apenas necessário seleccionar. Por defeito a função seleccionada é “Nenhuma Função”.

Figura 4.26 - Interface gráfica do menu Comandos RFID.

- **Código de barras:** Permite consultar a leitura capturada pelo leitor de código de barras. Além de permitir visualizar a leitura, permite também visualizar o código de barras associado à leitura. Para visualizar o código de barras, existe uma função que a partir dos 12 dígitos permite gerar o código de barras respectivo. Esta página Web, encontra-se constantemente em actualização, a uma taxa de 5 segundos, para actualizar a leitura capturada pelo *scanner*.

Figura 4.27 - Interface gráfica do menu Código de Barras.

- **Impressora:** Permite enviar comandos para realizar as funções na impressora, bem como inserir os dados, tais como, código de barras, texto e imagem. Na impressão de código de barras, após se ter inserido o respectivo código de barras, é possível visualizar o código associado. Na impressão de imagens, a imagem é carregada e deve estar em formato JPG. Para os comandos especiais, apenas é necessário seleccionar o comando a executar. Ainda neste menu, é possível visualizar os possíveis erros que

ocorram durante a impressão. Quando um comando é executado e ocorre um erro, este erro é indicado quando voltamos ao menu Impressora, logo no início da página Web.

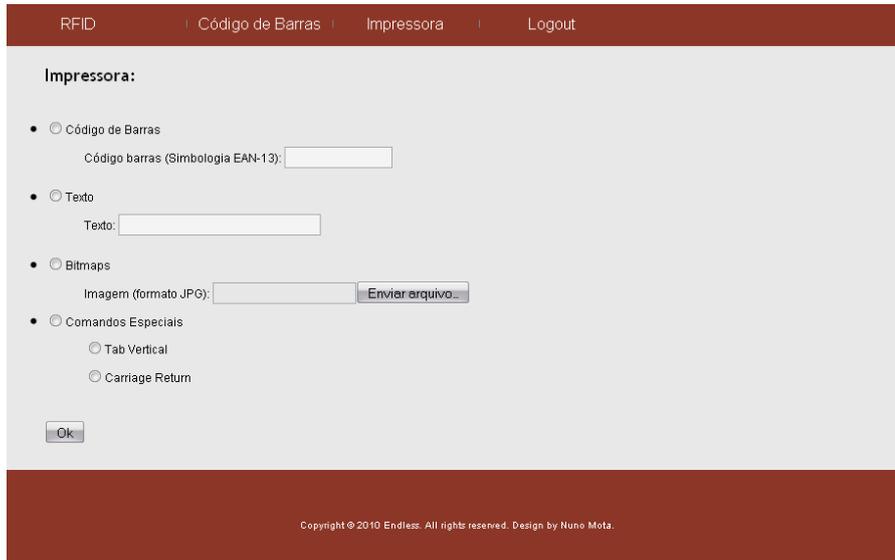


Figura 4.28 - Interface gráfica do menu Impressora.

- Login: Permite iniciar uma sessão.



Figura 4.29 - Interface gráfica do menu Login.

- Logout: Permite terminar uma sessão, e a página Web é redireccionada para a página Login.

Na Figura 4.40 apresenta-se, o diagrama de caso de usos (UML) que descreve as funcionalidades da aplicação Web. Um caso de uso, descreve uma sequência de acções que representa um cenário com o objectivo de demonstrar o comportamento de um sistema ou

parte dele a partir de interações. Os actores representam os dois tipos de utilizadores e os casos de uso os diferentes menus da aplicação Web.

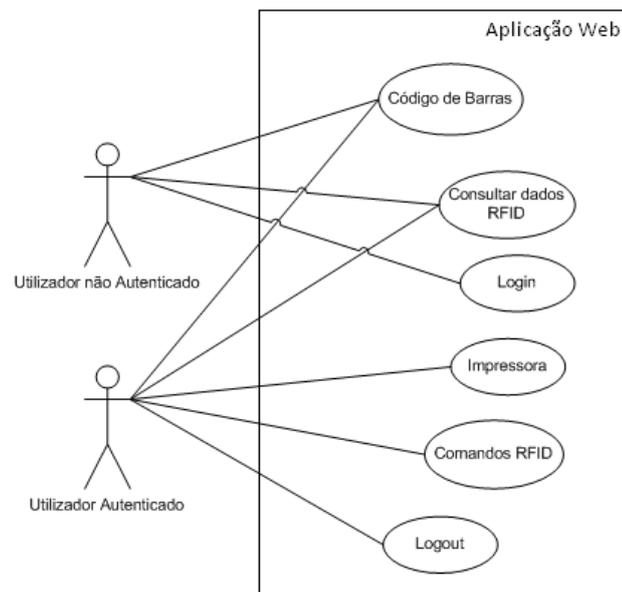


Figura 4.30 - Visão da interface Web utilizando casos de uso.

A interface Web desenvolvida apresenta uma grande potencialidade, pois permite aceder ao sistema de uma forma simples, utilizando um simples browser.

Capítulo 5

Validação da Solução Proposta

5.1 - Software Modbus Poll

Após terminada a implementação da aplicação procedeu-se à fase de testes. No presente capítulo serão apresentados alguns desses testes, para validação da solução implementada.

O *Software Modbus Poll*, foi o software utilizado para simular o cliente Modbus. A interface gráfica apresentada ao utilizador é a ilustrada na Figura 5.1, e é bastante simples e intuitiva. Para enviar os comandos Modbus, é necessário iniciar a ligação com o servidor Modbus, escolher a função, definir o endereço do primeiro registo e o número dos registos que se pretendem ler ou escrever.

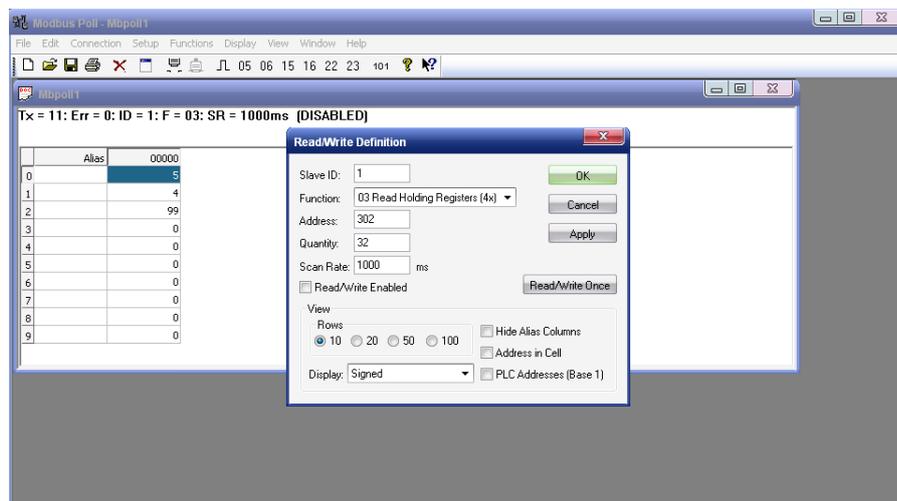


Figura 5.1 - Interface gráfica com o utilizador do *Software Modbus Poll*.

É conveniente referir alguns pontos, para uma melhor compreensão das figuras apresentadas. As iniciais 'TX', indicam o número de vezes de envio do comando. 'ERR', indica

o número de envios que resultam em erro. 'F', indica o código da função Modbus, e neste caso apenas será utilizado 03 (leitura) e o 16 (escrita). Os dados estão em base hexadecimal.

O primeiro passo, ilustrado na Figura 5.2, consiste em efectuar os pedidos das funções a realizar pelos diferentes leitores, que são efectuados no endereço 300. Como se pode ver, o leitor 1 não realizará nenhuma função, o leitor 2 realizará a função Inventory, o leitor 3 realizará a função de *Read Multiple Blocks*, e por fim o leitor 4 realizará a função de *Write Multiple Blocks* na etiqueta 1, indicado no endereço 301. O endereço 300 e 301, indicam também a ocorrência de erros, por exemplo, não existem etiquetas ao alcance do leitor ou etiqueta seleccionada para escrita incorrecta.

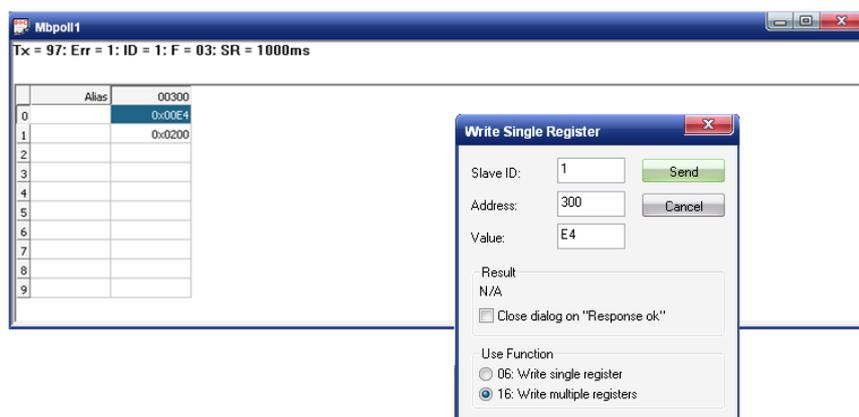


Figura 5.2 - Interface gráfica dos comandos enviados para os leitores RFID.

Na Figura 5.3, ilustra-se a leitura dos endereços 318 ao 333, que corresponde aos dados do leitor 2. Como se pode ver, os dados dos endereços 318 ao 325, correspondem aos UIDs das etiquetas ao alcance do leitor RFID, neste caso 2 etiquetas.

The screenshot shows the Mbpoll1 software interface displaying a table of data for reader 2. The table has columns for 'Alias' and 'Value'. The data is as follows:

	Alias	Value	Alias	Value	Alias	Value
0	00310	0xE7DD	00320	0x0000	00330	0x0000
1		0x5A29		0x0000		0x0000
2		0x07E0		0x0000		0x0000
3		0xAC80		0x0000		0x0000
4		0xE8DD				
5		0x6F2F				
6		0x0000				
7		0x0000				
8	0x07E0	0x0000				
9	0xAC80	0x0000				

Figura 5.3 - Interface gráfica com dados do leitor 2.

Na Figura 5.4, ilustra-se a leitura dos endereços 334 ao 349, que corresponde aos dados do leitor 3. Os dados apresentados nos endereços 346 ao 349, com tamanho de 4 bytes corresponde aos dados da etiqueta ao alcance do leitor.

	Alias	00330	Alias	00340
0				0x0000
1				0x0000
2				0x0000
3				0x0000
4		0x0000		0x0000
5		0x0000		0x0000
6		0x0000		0x3032
7		0x0000		0x3032
8		0x0000		0x0000
9		0x0000		0x0000

Figura 5.4 - Interface gráfica com dados do leitor 3.

Na Figura 5.5, ilustra-se a leitura dos endereços 350 a 365, que corresponde aos dados do leitor 4. A função a realizar é a *Write Multiple Blocks*, na etiqueta indicada nos endereços 350 a 353, dos dados nos endereços 362 a 365.

	Alias	00350	Alias	00360
0		0x07E0		0x0000
1		0xAC80		0x0000
2		0xE7DD		0x3131
3		0x5A29		0x3232
4		0x07E0		0x3333
5		0xAC80		0x3434
6		0xE8DD		
7		0x6F2F		
8		0x0000		
9		0x0000		

Figura 5.5 - Interface gráfica com dados do leitor 4.

Na Figura 5.6, ilustra-se a leitura dos endereços 366 a 372, que corresponde aos dados do leitor de código de barras.

	Alias	00360	Alias	00370
0				0x3534
1				0x3537
2				0x0035
3				
4				
5				
6		0x3635		
7		0x3130		
8		0x3133		
9		0x3032		

Figura 5.6 - Interface gráfica com dados do leitor código de barras.

Na Figura 5.7, ilustra-se a leitura dos endereços 373 ao 379, que corresponde à impressão do código de barras. Nos endereços 374 ao 379, encontra-se o código de barras a imprimir, e o endereço 373 corresponde ao comando impressão de código de barras.

Mbpoll1
Tx = 34: Err = 6: ID = 1: F = 03: SR = 1000ms

	Alias	00370
0		
1		
2		
3		0x0001
4		0x3231
5		0x3433
6		0x3635
7		0x3837
8		0x3039
9		0x3231

Figura 5.7 - Interface gráfica de impressão código de barras.

Na figura 5.8, ilustra-se a leitura dos endereços 373 ao 393. Nos endereços 374 ao 389, encontra-se o texto a imprimir, e o endereço 373 corresponde ao comando de impressão de texto.

Mbpoll1
Tx = 21: Err = 0: ID = 1: F = 03: SR = 1000ms

	Alias	00370	Alias	00380	Alias	00390
0				0x4520		0x0000
1				0x676E		0x0000
2				0x6E65		0x0000
3		0x0002		0x6168		0x0000
4		0x6146		0x6972		
5		0x7563		0x2061		
6		0x646C		0x6164		
7		0x6461		0x5520		
8		0x2065		0x502E		
9		0x6564		0x002E		

Figura 5.8 - Interface gráfica de impressão texto.

Na figura 5.9, ilustra-se a leitura dos endereços 373 ao 423. No endereço 373 indica-se o comando de impressão de imagem. Nos endereços 374 aos 423 encontra os dados relativos à imagem.

Tx = 322: Err = 0: ID = 1: F = 03: SR = 1000ms

	Alias	00370	Alias	00380	Alias	00390	Alias	00400	Alias	00410	Alias	00420
0				0x0000		0x0000		0x0000		0x0000		0x0303
1				0x0000								
2				0xFD0D		0x0000		0x0000		0x0000		0x0000
3				0x0303		0x0000		0x0000		0x0000		0x0101
4		0x0003		0x0000		0x0000		0x0000		0x0000		
5		0x0000		0x0303		0x0000		0x0000		0x0000		
6		0x0000										
7		0x0000		0x0000		0x0000		0x0000		0xFD0D		
8		0x0000		0x0101		0x0000		0x0000		0x0303		
9		0x0000										

Figura 5.9 - Interface gráfica de impressão imagem.

5.2 - Interface Web

Na Figura 5.10, ilustra-se a interface gráfica da aplicação Web. Como se pode ver, se o utilizador não se encontrar autenticado, apenas poderá consultar os dados dos diferentes dispositivos de recolha e identificação automática. Se estiver autenticado, poderá consultar os dados bem como enviar comandos para os diferentes dispositivos (Figura 5.11).



Figura 5.10 - Interface gráfica da aplicação Web, com utilizador não autenticado.



Figura 5.11 - Menu com utilizador autenticado.

Na Figura 5.12, ilustra-se a página Web que permite enviar os comandos, para executar as funções nos diferentes leitores RFID. Se a função a realizar é *Write Multiple Blocks*, torna-se necessário seleccionar a etiqueta onde se pretende escrever, bem como inserir os dados a escrever, tal como ilustrado, caso contrário os formulários não serão considerados válidos, e o respectivo pedido ao servidor não será efectuado. Isto corresponde à validação dos formulários do lado do cliente, tal como ilustrado na Figura 5.13.

Faculdade de Engenharia da Universidade do Porto
Linha de Fábrica Flexível no Laboratório de Controlo e Automação

RFID | Código de Barras | Impressora | Logout

Enviar comandos:

Leitor RFID 1:	Leitor RFID 2:
<input type="radio"/> Nenhuma função <input type="radio"/> Inventário <input type="radio"/> Leitura <input checked="" type="radio"/> Escrita <input checked="" type="radio"/> Etiqueta 1 <input type="radio"/> Etiqueta 2 <input type="radio"/> Etiqueta 3 Dados: <input type="text" value="12345678"/>	<input type="radio"/> Nenhuma função <input type="radio"/> Inventário <input checked="" type="radio"/> Leitura <input type="radio"/> Escrita <input type="radio"/> Etiqueta 1 <input type="radio"/> Etiqueta 2 <input type="radio"/> Etiqueta 3 Dados: <input type="text"/>
Leitor RFID 3:	Leitor RFID 4:
<input checked="" type="radio"/> Nenhuma função <input type="radio"/> Inventário <input type="radio"/> Leitura	<input checked="" type="radio"/> Nenhuma função <input type="radio"/> Inventário <input type="radio"/> Leitura

Figura 5.12 - Interface gráfica dos comandos dos leitores RFID.

A consulta de dados aos leitores RFID, é ilustrada nas Figuras 5.14 e 5.15. Inicialmente, deverá ser seleccionado o leitor ou leitores, os quais se pretende consultar os dados. Deverá ser seleccionado pelo menos um leitor, caso contrário não é efectuado nenhum pedido ao servidor (Figura 5.14). Permite visualizar os dados lidos, ou a ser escritos, os UIDs das etiquetas, bem como saber qual a função que os leitores RFID se encontram a executar e os respectivos erros, caso ocorram. Os dados visualizados encontram-se constantemente em actualização.

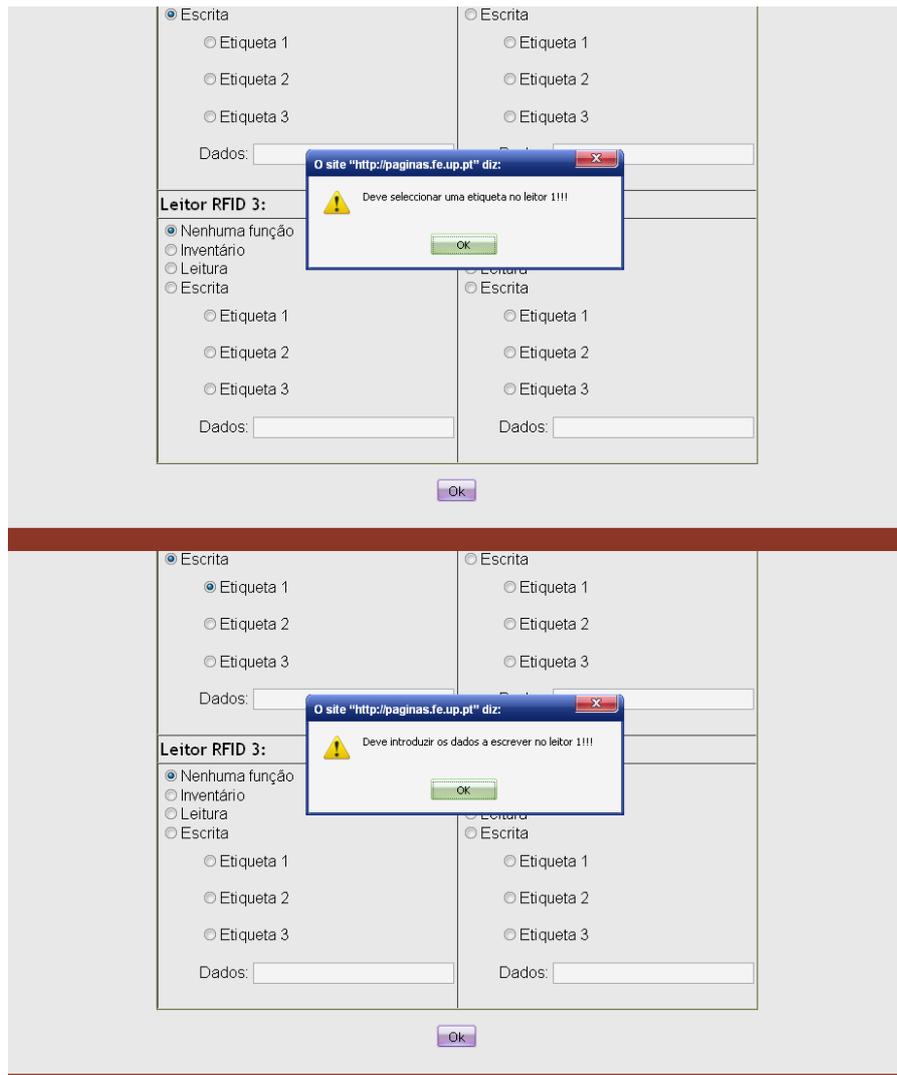


Figura 5.13 - Interfaces gráficas da validação dos formulários.



Figura 5.14 - Interface gráfica com escolha dos leitores RFID para consulta de dados e com validação do formulário.

Na figura 5.15, ilustra se a consulta de dados dos diferentes leitores, em diferentes instantes de tempo. É possível visualizar os diferentes dados, as funções que se encontram em execução e a ocorrência de erros.

Dados dos RFID's:	
• Dados leitor 1:	
Etiqueta 1 (Hex):	e00780acdde83376
Etiqueta 2 (Hex):	e00780acdde7295a
Etiqueta 3 (Hex):	e00780acdde82f6f
Dados:	12345678
Função a realizar:	Escrita
Status:	Sem erros.
• Dados leitor 2:	
Etiqueta 1 (Hex):	0000000000000000
Etiqueta 2 (Hex):	0000000000000000
Etiqueta 3 (Hex):	0000000000000000
Dados:	00000000
Função a realizar:	Leitura
Status:	Erro! Não existe tags ao alcance do leitor 2!
• Dados leitor 1:	
Etiqueta 1 (Hex):	e00780acdde83376
Etiqueta 2 (Hex):	e00780acdde7295a
Etiqueta 3 (Hex):	e00780acdde82f6f
Dados:	12345678
Função a realizar:	Escrita
Status:	Erro! Não existe tags ao alcance do leitor 1!
• Dados leitor 2:	
Etiqueta 1 (Hex):	0000000000000000
Etiqueta 2 (Hex):	0000000000000000
Etiqueta 3 (Hex):	0000000000000000
Dados:	12340000
Função a realizar:	Leitura
Status:	Sem erros.
• Dados leitor 2:	
Etiqueta 1 (Hex):	0000000000000000
Etiqueta 2 (Hex):	0000000000000000
Etiqueta 3 (Hex):	0000000000000000
Dados:	20000000
Função a realizar:	Leitura
Status:	Sem erros.
• Dados leitor 3:	
Etiqueta 1 (Hex):	e00780acdde83376
Etiqueta 2 (Hex):	0000000000000000
Etiqueta 3 (Hex):	0000000000000000
Dados:	00000000
Função a realizar:	Inventory
Status:	Erro! Não existe tags ao alcance do leitor 3!

Figura 5.15 - Interface gráfica da consulta de dados dos leitores RFID.

Na figura 5.16, ilustra-se a consulta da leitura realizada pelo leitor de código de barras. Para além de se visualizar a simbologia e a leitura na caixa de texto, é possível visualizar o código de barras associado. A leitura visualizada encontra-se constantemente em actualização.



Figura 5.16 - Interface gráfica de consulta da leitura realizada pelo leitor de código de barras.

Na figura 5.17, ilustra-se página Web com as funcionalidades disponíveis na impressora. Permite impressão de código de barras, texto, imagens e comandos não imprimíveis tais como *carriage return* ou *tab horizontal*. Na impressão de código de barras, após se ter inserido o código, é possível visualizar o código de barras antes de realizar a sua impressão, tal como ilustrado.

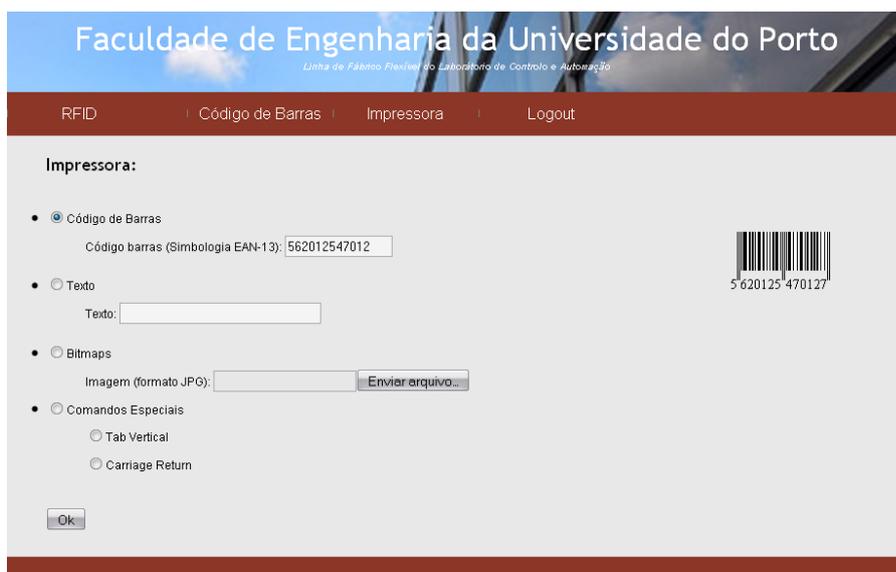
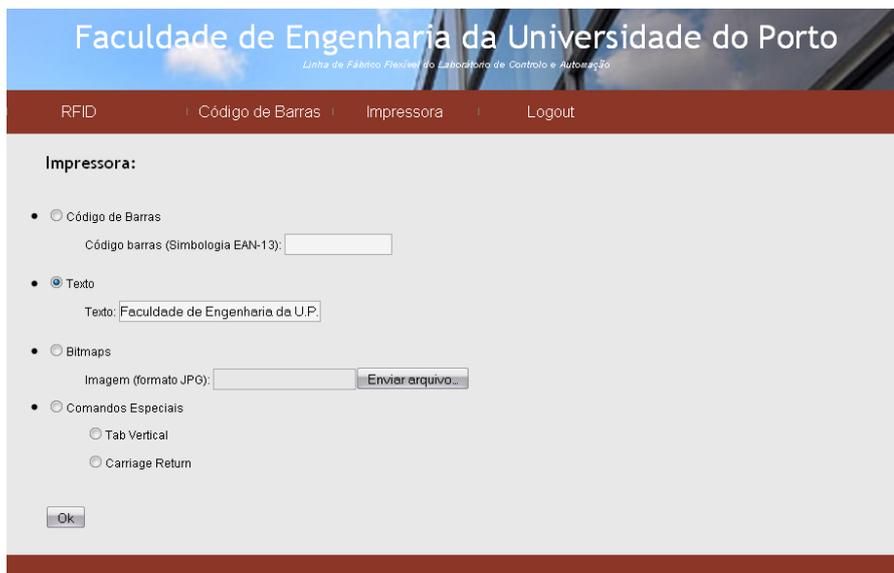


Figura 5.17 - Interface gráfica das funcionalidades disponíveis na impressora com visualização código de barras a imprimir.

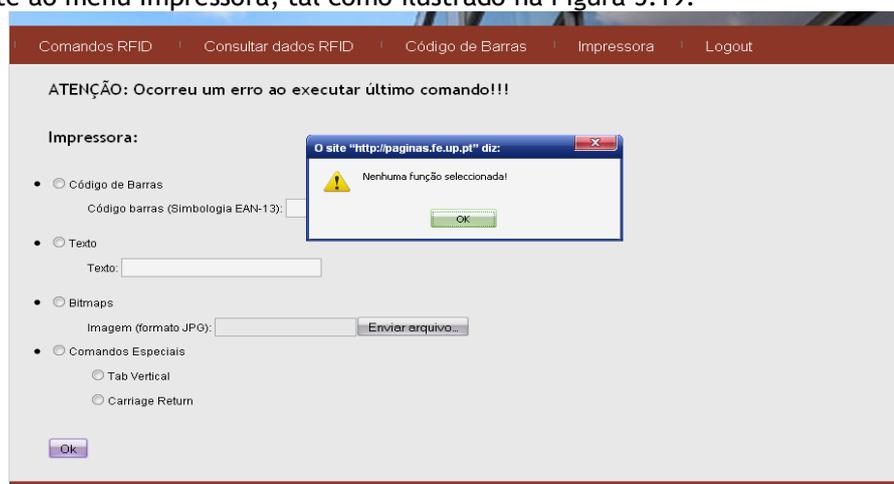
Para a impressão de texto, apenas é necessário introduzir o texto na respectiva caixa, tal como ilustrado na Figura 5.18. Na impressão da imagem, deve ser especificado o caminho onde a imagem se encontra, e esta deve estar em formato JPG. Quanto aos comandos especiais, torna-se apenas necessário seleccionar o comando pretendido. Também na página Web das funcionalidades da impressão, existe validação dos formulários do lado cliente, tal como ilustrado na Figura 5.19.



The screenshot shows a web interface for a printer. At the top, there is a header for 'Faculdade de Engenharia da Universidade do Porto' with a sub-header 'Linha de Fátima Freixas do Laboratório de Controlo e Automação'. Below this is a navigation bar with links for 'RFID', 'Código de Barras', 'Impressora', and 'Logout'. The main content area is titled 'Impressora:' and contains several radio button options: 'Código de Barras', 'Texto' (which is selected), 'Bitmaps', and 'Comandos Especiais'. Under 'Código de Barras', there is a text input field for 'Código barras (Simbologia EAN-13):'. Under 'Texto', there is a text input field containing 'Faculdade de Engenharia da U.P.'. Under 'Bitmaps', there is a text input field for 'Imagem (formato JPG):' and a button labeled 'Enviar arquivo...'. Under 'Comandos Especiais', there are two radio button options: 'Tab Vertical' and 'Carriage Return'. At the bottom left, there is an 'Ok' button.

Figura 5.18 - Interface gráfica com impressão de texto.

Na execução das funcionalidades disponíveis na impressora, podem ocorrer erros, tais como a introdução de caracteres não imprimíveis. A ocorrência de erros é indicada na página Web da impressora, mas apenas depois de ter realizado o pedido de impressão, e ter voltado novamente ao menu Impressora, tal como ilustrado na Figura 5.19.



The screenshot shows the same web interface as Figure 5.18, but with an error message displayed. At the top of the main content area, it says 'ATENÇÃO: Ocorreu um erro ao executar último comando!!!'. Below this, the 'Impressora:' section is visible, but the 'Texto' option is no longer selected. A dialog box is overlaid on the interface, titled 'O site "http://paginas.fe.up.pt" diz:', with a yellow warning icon and the message 'Nenhuma função seleccionada!'. There is an 'Ok' button in the dialog box. The 'Ok' button at the bottom left of the main interface is also visible.

Figura 5.19 - Interface gráfica das funcionalidades da impressora, com indicação de ocorrência de erros.

Na Figura 5.20, ilustra a interface gráfica apresentada sempre que os comandos são inseridos com sucesso. Isto significa que o pedido ao servidor foi realizado com sucesso, não eliminando a possibilidade de ocorrência de erros.



Figura 5.20 - Interface gráfica apresentada quando comandos inseridos com sucesso.

Capítulo 6

Conclusões e Futuros Desenvolvimentos

6.1 - Conclusões gerais

As tecnologias AIDC surgiram nos anos setenta, e estão hoje presentes em toda a parte. São utilizadas nas mais diversas áreas, constituindo um factor de competitividade no mundo dos negócios e das tecnologias, pois influenciam de forma importante as operações. As tecnologias AIDC podem ser classificadas em 6 categorias, *touch screen*, óptica, magnética, electromagnética e cartões inteligentes. A tecnologia óptica, particularmente os códigos de barras, é das tecnologias AIDC mais utilizada, devido à sua simplicidade e baixo custo, mas o seu futuro não é muito atractivo. No entanto a tecnologia electromagnética, particularmente o RFID, começa a ser muito utilizado, devido à maior capacidade de dados e robustez para ambientes adversos, e especialmente por não necessitar de estar visível para a sua leitura, e mais cedo ou mais tarde poderá ser a causa de “morte” dos códigos de barras. Estas duas tecnologias, RFID e código de barras, foram utilizadas para implementar um sistema para utilização na linha de fabrico flexível do laboratório de controlo e automação, que permitisse a sua utilização a partir do protocolo de comunicação Modbus.

O estudo dos protocolos de comunicação e dos diferentes modos de funcionamento, permitiu implementar na aplicação as funções de leitura e escrita dos dispositivos AIDC. O sistema foi desenvolvido para uma aplicação em concreto, mas poderá ser utilizado noutras aplicações semelhantes.

No servidor Modbus, definiu-se um conjunto de registos, bem como a sua organização, que permitisse implementar as funções pretendidas nos diferentes dispositivos. Assim, foram realizadas com êxito as funções de leitura e escrita numa etiqueta, função de reset aos leitores RFID, função para obter UID das etiquetas, função de leitura de códigos de barras e funções de impressão de código de barras, texto, imagens e alguns comandos não imprimíveis. Parte da aplicação encontra-se preparada para uma reorganização automática dos registos, caso se pretenda adicionar ou remover dispositivos AIDC no servidor

Modbus/TCP, ou alterar o número de blocos de dados a ler ou escrever nos diferentes dispositivos. A aplicação apresenta algumas limitações. A nível de configuração, não foram implementadas muitas opções, dando origem a uma solução pouco flexível, o que não constitui de todo uma desvantagem, pois não se pretende dar liberdade aos alunos para modificar os parâmetros de configuração, que possam provocar um mau funcionamento dos dispositivos AIDC. Desta forma, a maioria das configurações dos dispositivos encontram-se pré definidas. Quanto ao tratamento de erros por parte da aplicação, foi disponibilizado ao utilizador registos, onde estes são identificados e discriminados, revelando-se bastante importante. Para aceder e interagir com o sistema, foram criadas 2 interfaces, uma interface Web e a utilização de um software que suporta o protocolo Modbus, que simula um cliente Modbus/TCP. São 2 interfaces com potencialidades diferentes, que permite aos alunos aceder e “controlar” o sistema.

6.2 - Futuros Desenvolvimentos

Todos os objectivos propostos à priori foram atingidos, no entanto é possível fazer-se melhoramentos sobre o trabalho desenvolvido. Para a utilização específica da linha de fabrico flexível, pode-se então:

- Implementar restantes funções disponíveis nos leitores RFID.
- Implementar a configuração dos leitores RFID.
- Implementar as restantes configurações da impressora.
- Implementar os restantes modos de acesso à etiqueta.
- Acrescentar novas tecnologias AIDC.
- Melhorar a interface Web.

A aplicação pode ainda ser generalizada, para poder ser aplicada a sistemas que necessitem de tecnologias AIDC.

Referências

- [1] K. Michael, “Trends in the Selection Automation Identification Technology in Electronic Commerce Applications” in Trends in the selection of automatic identification technology in electronic commerce applications Chile: N. Cerpa & P. Bro 2003, pp135-152.
- [2] M.P. Groover, “Automation, production systems, and computer-integrated manufacturing”, Upper Saddle River: Pearson Prentice Hall, 2008.
- [3] Módulo de Impressão de Código de Barras, Disponível em: <http://www.bibliosoft.pt/suporte/docs/bb2004-mcb.pdf>. Acedido em Junho de 2010.
- [4] Identificação por Radiofrequência, Disponível em: http://www.oarquivo.com.br/portal/index.php?option=com_content&view=article&id=1950:refid-identificacao-por-radio-frequencia-parte-1&catid=71:ciencia-e-tecnologia&Itemid=368. Acedido em Junho de 2010.
- [5] Servidor Modbus/TCP para Sistemas de Identificação Automáticos (AIDC), José Filipe Alves Teixeira.
- [6] RFID World, Disponível em <http://www.rfidworld.com/>. Acedido em Maio de 2010.
- [7] RFID Technology, Disponível em: <http://rfidtek.blogspot.com/>. Acedido em Junho de 2010.
- [8] S. Prevadovic and N.C. Karmakar, “RFID Transponders - A Review”, in Electrical and Computer Engineering, 2006. ICECE06. International Conference on, 2006, pp. 96-99.
- [9] A.R. Solutions, Disponível em <http://www.atlasrfid.com/rfid-vs-barcode.asp>. Acedido em Maio de 2010.
- [10] Acerca do Reconhecimento Óptico de Caracteres, Disponível em <http://office.microsoft.com/pt-pt/help/HP030812552070.aspx>. Acedido em Junho de 2010.
- [11] Manual Leitor RFID, Disponível em: <ftp://labs-automacao.fe.up.pt/1005/manuais/RFID/OBID%20i-scan%20Reader/Systemmanual/H60301-3e-ID-B.pdf>. Acedido em Junho de 2010.
- [12] Manual da impressora Able Systems Limited Ap863-F, Disponível em: ftp://labs-automacao.fe.up.pt/1005/manuais/equipamento_controlo/impressora_termica/Able/Prog_Guide_1.3.PDF. Acedido em Junho de 2010.

- [13] PhpModbus, Disponível em: <http://code.google.com/p/phpmodbus/>. Acedido em Maio de 2010.
- [14] Understanding Radio Frequency Identification, Disponível em: <http://www.rmroz.com/rfid.html#f>. Acedido em Junho de 2010.
- [15] RFID Handbook - Fundamentals and Applications in Contactless Smart Cards and Identification (2nd Edition), pp 8.