



Universidade de Évora - Escola de Ciências e Tecnologia

Mestrado em Engenharia Mecatrónica

Dissertação

**Contributo para a Digitalização da Indústria do Mármore
com Identificação Automática por sistemas RFID**

Leonardo Henrique Sales de Andrade

Orientador(es) | João Manuel Figueiredo
Mouhaydine Tlemcani

Évora 2021



Universidade de Évora - Escola de Ciências e Tecnologia

Mestrado em Engenharia Mecatrónica

Dissertação

**Contributo para a Digitalização da Indústria do Mármore
com Identificação Automática por sistemas RFID**

Leonardo Henrique Sales de Andrade

Orientador(es) | João Manuel Figueiredo
Mouhaydine Tlemcani

Évora 2021



A dissertação foi objeto de apreciação e discussão pública pelo seguinte júri nomeado pelo Diretor da Escola de Ciências e Tecnologia:

Presidente | Fernando Manuel Janeiro (Universidade de Évora)

Vogais | João Manuel Figueiredo (Universidade de Évora) (Orientador)
Luís Filipe Soldado Granadeiro Rosado (Instituto Superior Técnico)

Para meus avos e meus pais

AGRADECIMENTOS

Aos meus pais, Sérgio e Elaine, aos meus avos paternos, Marçal e Luiza, e aos meus avos maternos, Manoel e Maria (in memoriam), por sempre terem sido meus maiores apoiadores e incentivadores no meu crescimento pessoal e minhas referências de vida.

Aos meus irmãos, Livia e Lucas, que mesmo à distância não nos impede de darmos boas risadas e conversas sérias.

À minha namorada, Manuela, que nos nossos momentos de stress, madrugadas de trabalho e pandemia estivemos firmes e fortes juntos, apoiando um ao outro.

Aos meus orientadores, João Figueiredo e Mouhaydine Tlemçani, por toda a paciência e conhecimento transmitidos nesse processo de trabalho árduo e nunca me permitiram desistir.

A Yanick, Fred, Gustavo e Roberto que foram amigos fundamentais durante meus anos em Portugal.

Aos meus antigos amigos que ficaram no Brasil ou estão espalhados pelo mundo, por nossas conversas quando nossas rotinas e fuso-horários permitem.

Aos meus amigos do laboratório que sempre foram bons em dar ideias e jogar conversa fora nos nossos intervalos de trabalho.

Ao Projeto BRO-CQ e ao Laboratório Instrumentação e Controlo pelo fornecimento dos dados e instrumentos utilizados na realização do presente trabalho.

Obrigado a todos !

Não tenhamos pressa, mas não percamos tempo.

José Saramago

RESUMO

Este trabalho visa contribuir com estratégias inovadoras a aplicar à cadeia produtiva da indústria do mármore, propondo novas abordagens tecnológicas, com a utilização de sistemas de Identificação por Radiofrequência - RFID. As capacidades dinâmicas das etiquetas RFID (read-write) permitem o acesso, através do UUI, ao armazenamento da caracterização física dos blocos de pedra, de acordo com os respectivos testes elétricos, ultrassônicos e visão de imagens tridimensionais. Estes parâmetros de caracterização dos blocos de pedra originais que resultam de ensaios não destrutivos, analisando a estrutura interna das rochas, podem ser armazenados em bases de dados, através da codificação permitida pelas etiquetas RFID, a fim de otimizar os subsequentes processos de corte e processamento. A tecnologia de identificação RFID, quando integrada numa rede de comunicação Ethernet, permite a comunicação automática com equipamentos de corte e processamento, constituindo uma plataforma industrial inteligente, integrando PCs (Computadores Pessoais) e PLCs (Controladores Lógicos Programáveis) no contexto da indústria 4.0. Além disso, outra grande vantagem da tecnologia RFID é que ela permite a rastreabilidade total do produto, permitindo que os consumidores finais invertam o processo produtivo

Resultados experimentais, para comprovar a eficácia da estratégia desenvolvida, são apresentados nesta dissertação.

Palavras Chaves: Identificação Automática; Sistemas RFID; Indústria 4.0; Indústria do Mármore

CONTRIBUTION TO THE DIGITIZATION OF THE MARBLE INDUSTRY WITH RFID IDENTIFICATION SYSTEMS

ABSTRACT

This work aims to improve the marble industry production chain by proposing new technological approaches using the Radio Frequency Identification - RFID - systems. The dynamic capabilities of the RFID (read-write) tags allow the storage of the physical characterization of stone blocks, through the UUI, according to the electrical, ultrasound and the three-dimensional image tests. These characterization parameters of the stone blocks, resulting from non-destructive tests, allow the knowledge of the internal structure of the rocks, which can be stored in databases, coded by RFID-tags in order to optimize their subsequent cutting and transformation processes. RFID identification technology when integrated into an ethernet communication network enables automatic communication with cutting and processing equipment, building an intelligent industrial platform, integrating PCs (Personal Computers) and PLCs (Programmable Logic Controllers) within Industry 4.0 environment. Additionally, another huge advantage of RFID technology is that it allows full product traceability, enabling end consumers to reverse the production path.

Experimental results, to prove the effectiveness of the developed strategy are presented in this dissertation.

KeyWords: Automatic Identification; RFID systems; Industry 4.0; Marble Industry

Lista de Figuras

1.1	Estrutura de um Sistema	8
1.2	Leitor Fixo em Portas	9
1.3	Leitor Portátil	10
1.4	Leitor em Empilhadeira	10
1.5	<i>Tag</i> Passivas	13
1.6	<i>Tag</i> Ativa	13
1.7	Smart-Label	14
1.8	Pedreiras em Vila Viçosa, Portugal	17
2.1	Ondas de Campo	19
2.2	Configuração de 4 Eléctrodos	20
2.3	Arranjos de Eléctrodos	21
2.4	Configuração do Arranjo Implementada	23
2.5	Dados Estruturados - Resistividades	24
2.6	Perfil de Resistividades	24
2.7	Ensaio ultrassónico	25
2.8	Dados Estruturados - Ultrassom	26
2.9	Imagem da Superfície - Binarização	29
2.10	Perfil de Descontinuidades	30
2.11	Dados Estruturados – Imagem	30
3.1	Ciclo de Vida do Mármore	31
3.2	Estrutura do Protótipo desenvolvido na Universidade de Évora	32
3.3	Ensaio Laboratoriais (esquema de comunicações)	33
3.4	Ensaio Laboratoriais (protótipo implementado na Universidade de Évora)	33
3.5	Protótipo de Produção Industrial (esquema de comunicações)	34
3.6	Protótipo de Produção Industrial (implementação na Universidade de Évora)	34
3.7	Fluxograma do Protocolo de Comunicação	35
3.8	Representação Gráfica da Aplicação do PC-APPLICATION	35
3.9	Representação Gráfica da Plataforma Web	36
3.10	Organização da Base de Dados <i>Online</i>	37
4.1	SOPAS - Tela inicial	38
4.2	SOPAS - Leitura rápida	39
4.3	SOPAS - Leitura/escrita de dados das Tags	39
4.4	SOPAS - Ajuste de alcance da antena	40
4.5	SOPAS - Temporização das aquisições	40

4.6	SOPAS - formatar output do leitor	41
4.7	SOPAS - Selecionar Formatar com UII da TAG	41
4.8	SOPAS - Formatar UII baseado na tabela ASCII	41
4.9	SOPAS - Endereço TCP/IP e modo de trabalho	42
4.10	SOPAS - Adicionar a fila de gravação novos parâmetros	42
4.11	SOPAS - gravar os novos parâmetros	43
4.12	SOPAS - Gravar os novos parâmetros de forma permanente	43
4.13	TIA portal - Rede Industrial	43
4.14	Fluxograma de trabalho do S7-1200	44
4.15	TIA PORTAL - TRCV_C	45
4.16	Fluxograma do Block 2	45
4.17	TIA PORTAL - BLOCK 2	46
4.18	Fluxograma do Block 3	46
4.19	TIA PORTAL - BLOCK 3	46
4.20	TIA PORTAL - BLOCK 1	47
4.21	TIA PORTAL - Tela da HMI e seus itens	47
4.22	TIA PORTAL - Função <i>SetBit</i> ao evento click	48
4.23	TIA PORTAL - Atribuição da <i>tag</i> ao campo de entrada/saída	48
4.24	SCADA - Painel de Controlo	49
4.25	SCADA - Interface PG/PC	49
4.26	SCADA - Diagnóstico PG/PC	49
4.27	SCADA - Conexões SCADA	49
4.28	SCADA - Lista de <i>scripts</i>	50
4.29	SCADA - Evento <i>value change</i>	50
4.30	SCADA - Aba eventos dos botões	50
4.31	SCADA - Selecionar <i>tag</i> para <i>inputs/outputs</i>	51
4.32	Ecran principal do SCADA	51
4.33	Fluxograma da Aplicação PC	52
4.34	PC local – Conexão do sensor RFID	53
4.35	PC local - Conexão com a Base Dados <i>online</i>	53
4.36	Terminal de entradas recebidas sensor RFID	53
4.37	PC local - Registo de novos objetos	53
4.38	Fluxograma da Aplicação <i>DATATRANSFER</i>	54
4.39	Ecrã Inicial da aplicação <i>DATATRANSFER</i>	54
4.40	Ecrã de conexão Base de Dados - <i>MySQL</i>	55
4.41	Ecrã de conexão Base de Dados - <i>SQL SERVER</i>	55
4.42	PLATAFORMA WEB – Visualização de scans dos sensores RFID	55

4.43	Plataforma WEB – Visualização de Objectos	55
4.44	Plataforma WEB – Vizualização de Áreas	56
4.45	PLATAFORMA WEB - EXIBIR/APAGAR NOVOS ARQUIVOS	56
4.46	PLATAFORMA WEB - BUSCA DE OBJECTOS	56
4.47	PLATAFORMA WEB - RESULTADO DA BUSCA	56
4.48	<i>SQL SERVER</i> - Painel de Controlo <i>ODBC</i>	57
4.49	<i>SQL SERVER</i> - Lista de Bases <i>ODBC</i>	57
4.50	<i>SQL SERVER</i> - Bases de dados com suporte <i>ODBC</i>	58
4.51	<i>SQL Server</i> -Dados do novo acesso	58
4.52	<i>SQL SERVER</i> - Autenticação do acesso à base	58
4.53	<i>SQL SERVER</i> - Escolha da base de dados	58
4.54	<i>SQL SERVER</i> - Dados do acesso ao <i>ODBC</i>	58
4.55	<i>SQL SERVER</i> - Status do acesso ao <i>ODBC</i>	58

Lista de tabelas

1.1	Classificações por Frequências	11
1.2	Classificação de tags EPC	13
1.3	Sumário comparativo entre os sistemas <i>Bar Code</i> e RFID	14
2.1	Resistividade dos Materiais	23

Lista de Abreviaturas

Auto-ID Tecnologias de Identificação Automática.

CPS Sistemas Ciber-Físicos.

EAN *European Article Number.*

EPC *Electronic Product Code.*

GPS *Global Positioning System .*

HMI *Human - Machine Interface.*

IoS *Internet dos Serviços.*

IoT *Internet das Coisas.*

IoTS *Internet das Coisas e Serviços.*

LAN *Local Area Network.*

LiDAR *Light Detection And Ranging.*

M2M Máquina - Máquina.

ODBC *Open Database Connection.*

PC *Personal Computer.*

PLC *Programmable Logic Controller.*

RFID Sistema de Identificação por Radio Frequência.

RTLS *Real Time Locating Systems.*

SCADA *Supervisory Control and Data Acquisition.*

SCL *Structured Control Language.*

SL *Smart Labels.*

SQL *Structured Query Language.*

TI *Tecnologies Information.*

UHF *Ultra - High - Frequency.*

UPC *Universal Product Code.*

UUI *Universal Unique Identifier.*

VBscript *Visual Basic Script.*

VS *Velocidade do Som.*

Índice

Lista de Figuras	XI
Lista de Tabelas	XIII
Lista de Abreviaturas	XV
1 Introdução	2
1.1 Indústria 4.0	3
1.1.1 Internet das Coisas e Serviços (IoTS)	5
1.1.2 Sistemas Ciber-Físicos (CPS)	6
1.1.3 Identificação e Localização Automática	7
1.2 Sistemas RFID	8
1.2.1 Tipos de Leitores	9
1.2.2 Tipos de Tags	10
1.2.3 RFIDs e Bar Codes	13
1.2.4 RFID e Sistemas de Localização em Tempo Real	14
1.3 Indústria do Mármore	15
1.3.1 Impactos Económicos	15
1.3.2 Impactos Ambientais	16
2 Digitalização da Indústria do Mármore	18
2.1 Ensaio Electro Resistivos	18
2.1.1 A relação entre a geologia e a resistividade	21
2.1.2 Processamento dos Dados	23
2.2 Ensaio de Ultrassons	24
2.2.1 Processamento dos Dados	26
2.3 Ensaio de Imagem	27
2.3.1 Processamento dos Dados	29
3 Conceção do Protótipo	31
4 Estratégia para Digitalização da Indústria do Mármore	38
4.1 Configuração dos Leitores RFID	38
4.1.1 Inicialização Rápida	38
4.1.2 Leitura/Escrita em Tags	39
4.1.3 Configurações da Antena	39
4.1.4 Configurações dos <i>Triggers</i> de Leituras	40
4.1.5 Formatação dos Dados de Saída	41

4.1.6	Configurações de Conexão - Ethernet	42
4.1.7	Gravação de Novos Parâmetros	42
4.2	Autômato Programável (PLC) e Painel de Operação (HMI)	43
4.2.1	Estrutura da Rede Industrial	43
4.2.2	Programação desenvolvida - S7 1200	44
4.2.3	Programação do Painel HMI	47
4.3	Sistema de Supervisão e Aquisição de Dados - SCADA	48
4.3.1	Configuração PG/PC	48
4.3.2	Configuração de SCRIPTS/TAGS/BOTÕES	49
4.4	Interfaces Gráficas - PC e DATATRANSFER	52
4.5	Plataforma Web	55
4.6	MYSQL e SQL SERVER	56
4.6.1	<i>MYSQL</i> - Estrutura dos dados	57
4.6.2	<i>SQL SERVER</i> - Configuração do ODBC	57
5	Conclusões e Sugestões para Trabalhos Futuros	59
A	ANEXOS	71
A.0.1	CODIGO DELPHI - RFID SYSTEM	72
A.0.2	CODIGO DELPHI - DATATRANSFER	75
A.0.3	PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO TRCV_C	78
A.0.4	PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO FIND	84
A.0.5	PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO MID	86
A.0.6	PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO LEFT	88
A.0.7	PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO CHARS_TO_STRG	90
A.0.8	CODIGO VBS - INSERT_SQL	93
A.0.9	CODIGO VBS - INSERT_OBJECT	94
A.0.10	MANUAL DE SINTAXE SQL	95

1 Introdução

O processo de industrialização começou com a introdução da manufatura mecânica no século 18, baseada nas máquinas a vapor, por James Watt [1]. A primeira revolução industrial foi uma enorme contribuição para a diminuição da fome nos países com desenvolvimento industrial, o que provocou a explosão populacional nestes países [2]. Este avanço foi seguido pela segunda revolução industrial que começou no século 20, com o surgimento da energia elétrica, da produção em massa de produtos, com base na divisão de trabalhos, principalmente devido à implementação das linhas de produção de Henry Ford [1]. Do início dos anos 70 até os dias atuais, seguiu-se o que foi chamada de terceira revolução industrial, caracterizada pelo desenvolvimento da eletrônica e das tecnologias de informação. O avanço tecnológico, a partir daí, teve um aumento substancial na automação dos processos industriais, o que resultou na substituição de parte significativa da mão de obra fabril por máquinas, tendo como consequência grandes transformações sócio-económicas e culturais. Por outro lado, o grande crescimento da produtividade, devido aos processos automáticos de fabrico, foram fundamentalmente devidos à normalização associada à produção em serie [2]. Por fim, a terceira revolução industrial, ainda está presente nos dias atuais, no entanto, verifica-se constantemente que a indústria se está a transformar com a digitalização dos processos e da informação, entrando numa nova fase de industrialização – A quarta revolução industrial ou Indústria 4.0.

A indústria de extração e transformação de rochas ornamentais em Portugal é uma das principais no mundo, com grande destaque no mármore. É na região do Alentejo que se situa uma das maiores reservas de mármore nacional, fonte de grande atividade exportadora. É muito comum durante o corte dos blocos de mármore acontecerem ruturas provocando enormes desperdícios e gerando significativos impactos económicos e ambientais. Deste modo, o projeto BRO-CQ [3] propõe utilizar ensaios não-destrutivos para auxiliar a tomada de decisão no momento do corte industrial. Dado que é fundamental que a fase dos ensaios laboratoriais e a fase industrial estejam interligadas, é fundamental o desenvolvimento da digitalização na indústria do mármore, não apenas interligando laboratórios e indústrias de extração/transformação, mas conectando todos os participantes da cadeia produtiva.

A presente dissertação tem como objetivos:

- Aproximar a indústria do mármore do conceito da indústria 4.0;
- Efetuar a digitalização da indústria do mármore, de modo a permitir a rastreabilidade dos seus produtos;
- Facilitar o armazenamento, controlo de *stocks*, transporte e venda aos consumidores finais;

-
- Integrar todos os participantes da cadeia produtiva numa base de dados *online*;

A presente dissertação encontra-se dividida em 5 capítulos. O primeiro capítulo descreve brevemente a indústria 4.0, destacando os principais conceitos dessa temática. Em sequência, é realizada uma introdução aos sistemas de identificação automática, principalmente os sistemas de identificação por radiofrequência (RFID). Por fim, é caracterizada a indústria do mármore e seus impactos económicos e ambientais. O segundo capítulo caracteriza os ensaios não-destrutivos em blocos de mármore, evidenciando os ensaios electro-resistivos, ultrassónicos, visão automática e respetivos processamentos de dados. O terceiro capítulo apresenta a conceção do protótipo experimental, descrevendo os instrumentos que foram utilizados, as ligações elétricas e as lógicas implementadas. Por fim, são apresentadas as aplicações desenvolvidas. O quarto capítulo descreve as implementações do sistema, nomeadamente os sensores RFID, a programação do PLC-S7 1200 e HMI, o sistema SCADA, as aplicações desenvolvidas em Delphi, a base de dados e a plataforma online. Por último, o quinto capítulo contém as conclusões finais e trabalhos futuros.

1.1 Indústria 4.0

Atualmente discute-se se o termo “revolução” é justificado para a mudança em andamento. Há quem afirme que uma definição mais razoável seria “evolução”, pois a transformação demorará várias décadas e os principais elementos que constituem este processo de transformação já existem e só serão desenvolvidos posteriormente. No entanto, pensamos que o termo “revolução” justifica-se, pois, a transformação tem características semelhantes a uma transformação de época, devido a uma mudança de paradigma na fabricação [4]. Uma definição é certa – será uma revolução/evolução para a digitalização. Em [5] destaca-se a mudança social em curso de uma sociedade industrializada para uma sociedade pós-industrializada baseada no conhecimento, orientada para serviços e baseada na informação, que é designada como revolução “digital”.

Devido a essas mudanças fundamentais, o governo alemão introduziu a “iniciativa estratégica” Indústria 4.0 em janeiro de 2011 [1]. Ela foi lançada pelo Grupo de Promotores de Comunicação da Aliança de Pesquisa em Ciência da Indústria (FU) [1]. As suas recomendações de implementação inicial foram formuladas pelo Grupo de Trabalho da Indústria 4.0 entre janeiro e outubro de 2012, sob a coordenação da Academia Nacional de Ciência e Engenharia (Acatech) [1]. Desde 2006, a Alemanha tem procurado uma estratégia de alta tecnologia para a coordenação de iniciativas de pesquisa e inovação, com o objetivo de garantir a sua forte posição competitiva industrial. O seu lançamento é conhecido como Estratégia de Alta Tecnologia 2020 e concentra-se em cinco áreas prioritárias: clima/energia, saúde/alimentação, mobilidade, segurança e comunicação [6]. Uma primeira e muito vaga definição do termo indústria 4.0 foi feita pela FU em 2011, em que se define indústria 4.0 como [1]:

“[...] a quarta revolução industrial, um novo nível de organização e controle de cadeias de valor inteiras ao longo de todo o ciclo de vida dos produtos. Este ciclo

inclui o atendimento aos requisitos individualizados do cliente e se estende desde a ideia, o pedido real, o desenvolvimento e a fabricação, a entrega ao cliente e o processo de reciclagem com os serviços envolvidos. A base para o desenvolvimento é formada pela disponibilidade de todas as informações necessárias em tempo real por meio da interconexão de todas as instâncias, que estão envolvidas na criação de valor, bem como pela capacidade de derivar o melhor fluxo de valor possível com base nos dados resultantes. Por meio da conexão de pessoas, objetos e sistemas, vão evoluir redes dinâmicas, otimizadas em tempo real, auto-organizadas e de valor cruzado, que podem ser otimizadas com base em diferentes critérios, como custos, disponibilidade e eficiência de recursos”

Outra definição foi trazida pela Acatech em 2013 [1], definindo a Indústria 4.0 como:

“... a integração técnica do CPS na manufatura e logística e o uso da Internet das Coisas e Serviços nos processos industriais. Isso terá implicações para a criação de valor, modelos de negócios, serviços posteriores e organização do trabalho.”

Atualmente, quase todas as empresas relacionadas com a tecnologia tentam encontrar uma explicação própria do que realmente constitui o conceito de Indústria 4.0. Isso torna ainda mais difícil, especialmente para as empresas, lidar com a complexidade deste tópico, já que os conteúdos essenciais são interpretados de modo diferente. De acordo com [7], a indústria 4.0 apresenta quatro componentes principais: Sistemas Ciber-Físicos (CPS), Internet das Coisas (IoT), Internet dos Serviços (IoS) e Fábricas Inteligentes. Tecnologias como comunicação máquina-máquina (M2M) e produtos inteligentes não são considerados componentes independentes da Indústria 4.0, já que a comunicação M2M é um habilitador da IoT e produtos inteligentes são um subcomponente dos CPS. Os autores percebem que *Big Data* e *Cloud Computing* são serviços de dados, que utilizam dados gerados em implementações da Indústria 4.0, mas não como componentes independentes da indústria 4.0 .

Essas considerações enfatizam a forte relação entre as principais tecnologias CPS, a Internet das Coisas e Serviços, como fundamento de todos os desenvolvimentos e consequências, que estão relacionados com o conceito da Indústria 4.0. Além disso, coloca-se o modelo da Fábrica Inteligente no centro de todas as reflexões futuras neste contexto.

A indústria 4.0 não deve ser abordada como um sistema fechado, mas sim como uma parte essencial de várias áreas-chave. Num mundo inteligente e interconectado, baseado na Internet das Coisas e Serviços, os principais setores económicos serão transformados em infraestruturas inteligentes. Essa transformação leva ao surgimento de redes e edifícios inteligentes no campo do fornecimento de energia, soluções de mobilidade e logística, inteligentes e sustentáveis, saúde inteligente e assim por diante. Portanto, a indústria 4.0 deve ser pensada, implementada e vivida de maneira interdisciplinar e em estreita cooperação com as outras áreas-chave [1]. Dentro desse ecossistema inteligente, a indústria 4.0 é a manifestação da abordagem do “pensamento inteligente” nos ambientes de produção.

No centro da Indústria 4.0, o conceito de Fábrica Inteligente constitui uma caracte-

terística chave. De acordo com especialistas, a *Smart Factory* possui várias características inovadoras, o que permite às empresas lidar com a complexidade e interrupções inesperadas, bem como fabricar produtos com maior eficiência e fiabilidade. Numa fábrica inteligente, pessoas, máquinas e recursos comunicam entre si tão naturalmente como numa rede social de humanos [1].

Para a elaboração e implementação de soluções da Indústria 4.0, o foco é definido nos três seguintes recursos abrangentes que simultaneamente priorizam áreas de ação [1] :

- **Integração horizontal:** Integração de vários sistemas de TI usados nas diferentes fases dos processos de fabricação e planeamento de negócios que envolvem uma troca de materiais, energia e informação dentro de uma empresa (por exemplo, logística de entrada, produção, logística de saída, marketing) e entre várias empresas diferentes (redes de valor).
- **Integração vertical:** Sistemas de TI apropriados, que podem fornecer suporte a toda a cadeia de valor, desde o desenvolvimento do produto até a engenharia do sistema de produção e serviços. É necessária uma abordagem holística de engenharia de sistemas, que inclui diferentes disciplinas técnicas.
- **Integração vertical em sistemas de produção em rede:** Integração de vários sistemas de TI em diferentes níveis hierárquicos dentro de uma empresa (por exemplo atuador e sensor, controle, gerenciamento de produção, fabricação e execução e níveis de planeamento corporativo) para fornecer uma solução integrada. O objeto de reflexão é a Fábrica Inteligente, de onde vêm esses sistemas e tecnologias em uso.

Essas três características principais do processo de transformação terão implicações de longo alcance, não apenas nos níveis económicos, na forma de novas oportunidades de negócios e novos modelos de negócios, mas também em termos de novas infraestruturas sociais, estruturas organizacionais e questões jurídicas. O espectro de tarefas para os funcionários, a qualificação necessária e a interação inovadora na forma de ampla cooperação homem-máquina terão um impacto significativo nos sistemas de trabalho socio-económicos futuros. As cadeias de valor interconectadas mudarão os modelos de negócios e organizações tradicionais que passarão de participantes competitivos no mercado a concorrentes cooperativos (“Coopetição”) [6].

1.1.1 Internet das Coisas e Serviços (IoTS)

A Internet das Coisas e Serviços surgiu englobando um número crescente de objetos inteligentes heterogêneos que se estão a vulgarizar na nossa atividade quotidiana [8]. Segundo [9] destaca-se a agregação de várias tecnologias que juntas formam a IoTS, na medida em que se apoiam mutuamente. Ao tentar entender o conceito de IoTS, é útil separar os termos “Internet das coisas” (IoT) e “Internet dos Serviços” (IoS) e explorar ambas as definições de per si.

Segundo [1] define-se IoT como:

“[...] vinculação de objetos físicos (coisas) com uma representação virtual na

internet ou numa estrutura semelhante à internet. A identificação automática por meio da tecnologia RFID é uma expressão possível da Internet das Coisas; por meio da tecnologia de sensor e atuador, a funcionalidade pode ser estendida pela detecção de status e execução de ações”.

Também segundo [1] define-se IoS como:

“[...] parte da internet, que fornece serviços e funcionalidades como componentes de software granulares baseados na web. [...]. As empresas podem organizar componentes de software individuais para soluções complexas, mas ainda flexíveis (com base em arquiteturas orientadas para serviços) [...]”

A transição da atual *Internet* “orientada para o homem” para a *internet* “orientada para as coisas” já está em vigor, e a linha de separação entre o mundo real e o virtual está a desvanecer-se rapidamente [10]. A IoTS é considerada como o elemento habilitador que irá integrar e interligar diferentes setores industriais. A possibilidade de tornar entidades aleatórias identificáveis e localizáveis (por exemplo, consumíveis, roupas, eletrodomésticos, máquinas, meios de transporte, plantas, animais, pessoas, etc.), independentemente da localização das coisas interligadas, abre oportunidades de negócios totalmente novas [11]. Todo o potencial e os resultados de longo prazo da IoTS não são completamente previsíveis, como aconteceu com a *internet* na década de 1960. Consequentemente uma ampla gama de investigadores na indústria e na academia, bem como as empresas e agências governamentais estão a demonstrar um interesse crescente no desenvolvimento da IoTS [10].

1.1.2 Sistemas Ciber-Físicos (CPS)

Deve referir-se que subjacente ao conceito da Indústria 4.0 não existe uma nova e única “Tecnologia da Indústria 4.0” [12], mas sim o progresso continuado da tecnologia da informação e comunicação em combinação com um crescimento exponencial da capacidade de computação, transmissão e armazenamento, que permite o surgimento de novos sistemas tecnológicos cada vez mais poderosos e interconectados [13], denominados de Sistemas Ciber-Físicos (CPS), conjuntamente com algumas tecnologias novas, ou já existentes, mas com novas aplicações [12]. Na literatura podem encontrar-se inúmeras definições de CPS. De acordo com [14] CPS são definidos como:

“[...] integrações de meios virtuais com processos físicos. Computadores e redes, interconectados, supervisionam e controlam os processos físicos, geralmente com ciclos de feedback, onde os processos físicos afetam os dados e vice-versa.”

O Comitê Alemão de Especialistas na Indústria 4.0 define o CPS da seguinte forma:

“Sistemas que vinculam diretamente objetos e processos reais (físicos) com objetos e processos de dados/informação (virtuais) por meio de redes de informáticas abertas, parcialmente globais e sempre interconectadas” [15].

Mesmo que não haja uma distinção nítida do que constitui um CPS, a principal

característica adicional desses sistemas pode ser vista muito simplesmente na ligação direta (combinação inteligente) dos mundos “real” e “virtual” [13]. Deste modo, o acoplamento de componentes de processamento de informação e objetos físicos na automação não é novo e já existe desde os anos 1970. A diferença essencial, é a interconexão de objetos e processos por meio de redes de informação abertas e globais, com recurso à *internet* [16]. O principal motor tecnológico para o surgimento do CPS tem origens baseadas em *hardware* e *software*. Por um lado, a infraestrutura tecnológica consistente de sistemas embebidos, sensores de alto desempenho, atuadores e interfaces de comunicação fornecem as capacidades de *hardware* necessárias ao contexto [13]. A maioria desses sistemas embebidos são atualmente “caixas” fechadas que não expõem a capacidade de computação ao exterior [14]. Por outro lado, o uso da web de negócios, plataformas de integração e serviços baseados em soluções em nuvem abre novas oportunidades de negócio [13]. Outra característica que pode ser utilizada para descrever um CPS e simultaneamente evidenciar as diversas dimensões desses sistemas, é o grau de descentralização de sua estrutura e do seu volume espacial. Através de tecnologias avançadas de micro sistemas, um CPS pode ser colocado num único *microchip*, incluindo vários sensores e um microprocessador para um processamento de dados. Um CPS maior pode ser construído na forma de uma máquina-ferramenta inteira, que por sua vez pode ser parte de um CPS ainda maior, uma fábrica inteira. Uma manifestação extrema de um CPS seria sua alocação de uma rede mundial, por ex. uma empresa com operação mundial [17].

1.1.3 Identificação e Localização Automática

De entre todas as elaborações sobre as funções dos Objetos Inteligentes, uma identidade única é percebida como uma funcionalidade central, que constitui a natureza mais fundamental de um Objeto Inteligente [8]. De acordo com a norma DIN 6763, “identificação” inclui um “[...] reconhecimento único e inconfundível de um objeto com base em características essenciais (características de identificação) com uma precisão predefinida em relação à finalidade pretendida” [18].

Uma identificação automática, na verdade, é o elo conectivo entre o fluxo de material e o fluxo de informação - o momento em que o mundo físico e virtual se intersectam [18]. Atualmente, no contexto industrial automatizado, a identificação automática por meio de tecnologias de identificação automática (*Auto-ID* ou AIT) prevalece [19]. Em [20], são definidas três dimensões principais das tecnologias de *Auto-ID*, que incluem identificação óptica, identificação de rádio e identificação de rede. A identificação óptica inclui principalmente o uso de sistemas de código de barras. Esses sistemas são o padrão mundial para identificação automática e, portanto, percebidos como a mais importante tecnologia de *Auto-ID*. O código de barras de um item pode ser impresso como código 1D (por exemplo, traços) ou código 2D (por exemplo, matriz) e é reconhecido por meio de um sensor óptico (câmara de vídeo ou feixe de laser [21]). Os dois padrões mais comuns são o *Universal Product Code* (UPC) e o *European Article Number* (EAN) [22]. As desvantagens dos sistemas de código de barras prendem-se com o fluxo lento de informação, uma vez que os códigos de barras são uma tecnologia de linha de visão que requer leitura sequencial item a item [23]. A tecnologia de Identificação

por Radiofrequência (RFID) supera essa significativa limitação tornando-se deste modo um factor decisivo para a realização de IoTS [11].

1.2 Sistemas RFID

Os sistemas de identificação por radiofrequência (RFID) estão inclusos nas tecnologias de identificação automática - AITs que cresceram constantemente durante o último meio século até hoje, e atualmente são indispensáveis na vida quotidiana (alimentação, medicamentos, vestuário, etc.). Porém o uso de RFIDs não está associado simplesmente à substituição dos *Bar Codes*, mas fundamentalmente às suas características dinâmicas de armazenamento de dados, que permitem auxiliar no planeamento de prazos e interoperabilidade/ sincronização de dados [24]. Os sistemas RFID estão a ser constantemente disseminados na maioria dos domínios tecnológicos, onde a integridade e a fiabilidade dos dados são fatores fundamentais de negócio. De fato a utilização de tags RFID pode ser visível em diversos sectores, nomeadamente: Produção industrial [25], Sector Aeroespacial [26], Retalho [27], Saúde [28], Logística [29], Defesa [30].

De acordo com [24, 31] os sistemas RFID são compostos por três componentes: *Tags*, *Reader* e *Host*, conforme ilustrado na Fig. 1.1.

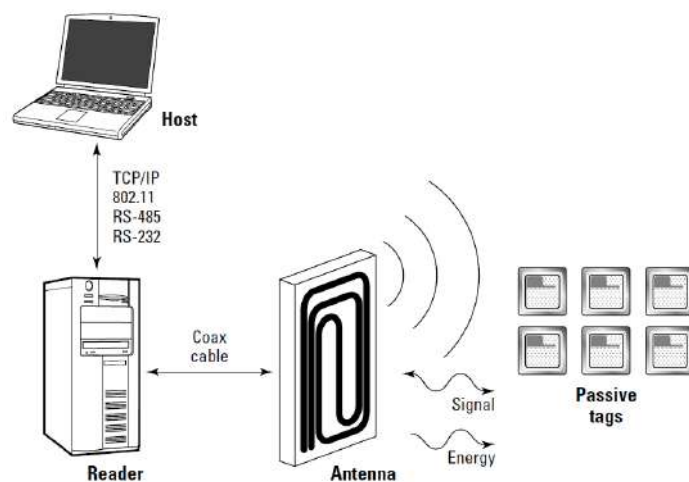


Figura 1.1: Estrutura de um Sistema RFID (Source: [24])

1. As *tags*, também chamadas de *transponders* são basicamente compostas por uma antena, um circuito integrado que armazena as informações de objetos ou pessoas (as *tags* podem, ou não, apresentar baterias);
2. Os *readers* são leitores responsáveis por enviar sinais no seu raio de alcance, e caso sejam detetadas *tags*, realizam a leitura das informações presentes. Constituem-se por uma antena, um módulo de radiofrequência e um módulo de controlo interno;
3. Como *Host*, ou controlador do sistema, geralmente utiliza-se um computador que executa o software de controlo/gestão do processo.

1.2.1 Tipos de Leitores

Os leitores são classificados de acordo com o seu tipo de operação de leitura ou de leitura/escrita. Os leitores de apenas leitura somente emitem ondas de sinal, e quando detetam uma *tag* no seu campo de trabalho, realizam a leitura dos dados armazenados. Os leitores de leitura-escrita além de emitir sinais e ler as *tags*, podem também escrever novos dados nas memórias internas das *tags* [32]. Outra classificação importante para os leitores é relativamente à sua mobilidade:

- **Leitores Fixos:** São fixados em locais específicos, que geralmente esperam que os itens passem, como por exemplo em transportadoras, portas de docas e retalhistas (Fig. 1.2). Qualquer item que passe no alcance dos sensores é detetado e a informação lida é anexada ao sistema. A vantagem de um sensor fixo é que as *tags* são lidas automaticamente e a desvantagem seria a instalação em lugares inadequados que impossibilitaria a leitura de forma otimizada. Fatores como temperatura, humidade, vibrações e proximidade de materiais metálicos, são desafios a considerar na escolha de leitores fixos;



Figura 1.2: Leitor Fixo em Portas (Source: [33])

- **Leitores Portáteis:** Possuem todos os elementos básicos de um leitor, incluindo antena e *software*. A informação recolhida das *tags* são armazenadas no próprio leitor e depois transferidas para o sistema. Os leitores portáteis oferecem maior flexibilidade, já que o utilizador pode transportar o leitor para próximo dos itens a controlar (Fig. 1.3). Quando comparados com os leitores fixos, estes são classificados para leitura/escrita em curto alcance e são frequentemente aplicados em ambientes médicos e escritórios;



Figura 1.3: Leitor Portátil (Source: [34])

- **Leitores Montados em Veículos:** São vulgarmente fixados em veículos como empilhadores e por isso podem ser chamados de “fixos-móveis” (Fig. 1.4) . Cobrem uma área maior quando comparados com os fixos e possuem um alcance maior que os portáteis. Por estarem posicionados num veículo, podem instalar-se todos os componentes do sistema RFID (leitores e *Host* para comunicação com os servidores). O único desafio enfrentado pelos leitores nesta aplicação é a proximidade de componentes metálicos que podem refletir o sinal de radiofrequência.



Figura 1.4: Leitor em Empilhadeira (Source: [35])

1.2.2 Tipos de Tags

As *tags* possuem classificações por frequências de trabalho, por tipos de alimentação e por classes de acordo como o seu EPC (*Electronic Product Code*). De acordo com a tabela 1.1 , as *tags* apresentam as seguintes frequências de trabalho e suas respectivas aplicações [36].

Tabela 1.1: Classificações por Frequências

Frequência	Alcance	Aplicações
(125 - 134) kHz	Baixa - Frequência	Identificação de Animais
13.56 MHz	Alta - Frequência	Acesso e Segurança
(433 - 956) MHz	Ultra - Alta - Frequência	Logística
2.45 GHz	Ultra - Alta - Frequência	Comunicações Móveis
5. 9 GHz	Super - Alta - Frequência	Sistemas de Transportes Inteligentes

Conforme relatado por [32, 37, 38], ao classificar tags pelo seu tipo de alimentação, obtemos três categorias: passivas, semi-ativas e ativas.

- **Tags Passivas:** As *tags* passivas não apresentam alimentação de energia, como baterias, logo não iniciam comunicação com o leitor RFID, então o sinal do leitor “acorda” a *tag* e a mesma responde. O processo ocorre com os seguintes passos:
 1. A antena da *tag* passiva recebe o sinal do leitor;
 2. O sinal é enviado para o circuito integrado da *tag*;
 3. Parte do sinal é utilizado para alimentar o circuito integrado;
 4. Por fim, o circuito integrado processa o sinal e responde ao leitor.

Geralmente as *tags* passivas apresentam as seguintes características:

- **Posicionamento:** A *tag* passiva depende da potência do seu leitor, portanto para obter energia precisa estar na “zona de interrogação” para gerar energia e responder;
 - **Tamanho e Alcance:** Como não há autoalimentação, as *tags* passivas tendem a ter menores dimensões;
 - **Vida Útil:** Por não haver autoalimentação, não há a necessidade de substituição da *tag* em si, apresentando assim grande longevidade;
 - **Mémoria:** A capacidade de memória das *tags* passivas varia de 1 bit a vários kbytes.
- **Tags semi-ativas:** As *tags* semi-ativas possuem a sua própria alimentação, como uma bateria, porém não iniciam a comunicação, respondendo apenas ao leitor quando recebem o sinal de “acordar”. As características comuns das *tags* semi-ativas são:
 - **Operação:** As *tags* semi-ativas respondem apenas se o sinal do leitor for recebido, tal como as *tags* passivas;
 - **Tamanho e Alcance:** Apresentam uma maior dimensão relativamente às *tags* passivas devido à existência de bateria e conseqüentemente também um alcance de transmissão maior;

-
- **Vida Útil:** A vida útil das *tags* semi-ativas é mais curta que as passivas, devido à longevidade da bateria utilizada;
 - **Mémoria:** A capacidade de memória varia, porém geralmente é maior que as *tags* passivas, por apresentarem um tamanho maior, mais espaço para componentes e bateria de alimentação.
 - **Tags Ativas:** As *tags* ativas possuem autoalimentação (bateria) e podem iniciar a comunicação emitindo seu próprio sinal para o leitor, o que significa que diferentemente das *tags* passivas e semi ativas não precisam de ser “acordadas” pelo leitor, para geração de energia para alimentar o circuito interno e sinal de resposta. Seguidamente são enumeradas as características das *tags* ativas:
 - **Operação:** Por apresentarem autoalimentação, são possíveis 2 modos de trabalho. No primeiro, a *tag* funciona permanentemente enviando dados. No segundo modo a *tag* espera receber o sinal do leitor. Importa ressaltar que a *tag*, em modo de trabalho permanente, pode também enviar sua localização a intervalos de tempo determinados;
 - **Tamanho e Alcance:** Devido às baterias, *tags* ativas têm maior dimensão quando comparadas com outras *tags*. No entanto, com o avanço tecnológico, as *tags* estão reduzindo a sua dimensão. Algumas *tags* ativas apresentam alcances aproximados de 1 km, contudo, confinadas dentro dos padrões e regulamentos, muitas *tags* apresentam alcance de 10 m. Devido seu longo alcance é muito comum serem integradas com o *Global Positioning System* (GPS) para ter sua exata localização;
 - **Vida Útil:** Depende da bateria, porém geralmente têm duração de 10 anos;
 - **Mémoria:** A capacidade de memória é geralmente maior relativamente às *tags* semi-ativas e passivas, principalmente devido à maior dimensão e bateria;
 - **Transponders Activos:** Estas *tags* ficam apenas ativas quando recebem o sinal do leitor, o que possibilita um prolongamento da sua bateria;
 - **Beacon:** *Beacon* é designada uma *tag* que emite sinais em intervalos de tempo pré-determinados. Geralmente empregados em *Real-Time Locating Systems* (RTLS) que são adequadamente aplicados em rastreamento de peças em grandes instalações industriais e em operações de resgate da marinha e aeronáutica.

Finalmente as *tags* podem ser classificadas de acordo com o seu *Electronic Product Code* (EPC) que é a sintaxe para identificadores únicos designado para objetos físicos, unidades de trabalho, localizações e outras entidades identificadas dentro das operações [39]. A tabela 1.2 apresenta a classificação EPC:

Tabela 1.2: Classificação de tags EPC

Classe da Tag	Tipo da Tag	Mémoria	Comunicação
Classe 0	Passiva	Apenas Leitura	Não inicia comunicação
Classe 0+	Passiva	Igual a classe 0, mas permite uma escrita	Não inicia comunicação
Classe 1	Passiva	Leitura e Escrita Única	Não inicia comunicação
Classe 2	Passiva	Leitura e Escrita Única	Não inicia comunicação
Classe 3	Semi Ativa	Leitura e Reescrita	Não inicia comunicação
Classe 4	Ativa	Leitura e Reescrita	Inicia comunicação; Possui bateria; Permite comunicação entre tags
Classe 5	Ativa	Leitura e Reescrita	Inicia comunicação; Possui bateria; Permite comunicação entre tags

Abaixo seguem figuras de alguns tipos de tags citadas:

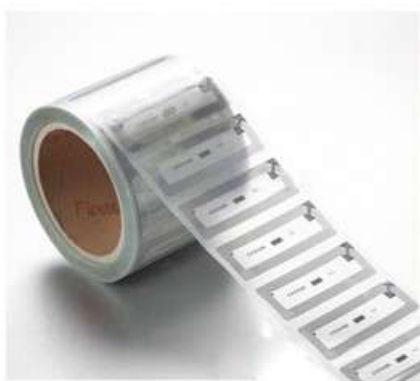


Figura 1.5: *Tag* Passivas (Source: [40])

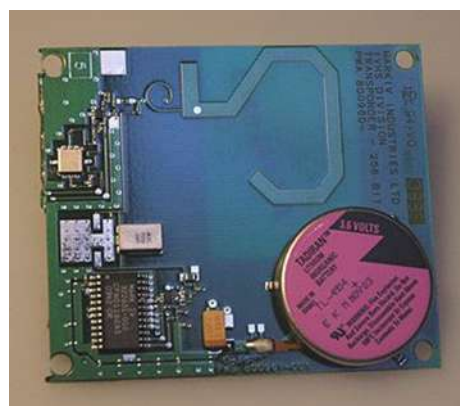


Figura 1.6: *Tag* Ativa (Source: [41])

1.2.3 RFIDs e Bar Codes

Antes da chegada dos sistemas RFID, a tecnologia de *Bar Codes* era a mais utilizada e até hoje continua sendo o maior concorrente em tecnologias de identificação. Em [31,33] é possível verificar as vantagens e desvantagens entre os sistemas RFID e *Bar Code*.

- **Vantagens:**

- Redução drástica da carga de trabalho, visto que os sistemas de *Bar Code* apenas realizam leituras item a item, enquanto os sistemas RFID permitem leituras múltiplas;
- Os sistemas RFID permitem leituras a distâncias mais longas, enquanto que os sistemas de *Bar Code* necessitam de contato visual;
- Os sistemas RFID permitem ler e escrever novos dados, atualizando os conteúdos armazenados nas etiquetas, enquanto os *Bar Codes* só podem ler e armazenar dados estáticos.

- **Desvantagens:**

- Os sistemas RFID são significativamente mais dispendiosos que os sistemas de *Bar Code*, especialmente quando se utilizam *tags* ativas.

Segundo [32], pode dizer-se que as tecnologias dos sistemas RFID preenchem os limites de operação dos sistemas de *Bar Codes*, considerando atualmente o surgimento dos *Smart Labels* (SL). Os SLs apresentam a inteligência e funcionalidade de uma *tag* RFID e a conveniência dos *Bar Codes*, por poderem ser facilmente impressos (Fig. 1.7). Os *Bar Codes* estão presentes na generalidade dos produtos que compramos e auxiliam os retalhistas e fornecedores a rastrear os seus stocks. Esses códigos são como impressões digitais para os produtos. Conforme já atrás referido, as *tags* RFID podem alargar consideravelmente as características dos *Bar Codes*. Por fim a utilização de SLs acaba por unir as vantagens dos dois sistemas.



Figura 1.7: Smart-Label (Source: [42])

A maior barreira na penetração de mercado dos sistemas RFID, é o custo das *tags*, dado que o custo dos *Bar Codes* é abaixo de 0.01/unidade enquanto que o custo das *tags* passivas são significativamente mais dispendiosas. No entanto a sua utilização justifica-se devido ao aumento da eficiência das operações relativamente aos *Bar Codes* [31]. Na tabela 1.3, apresenta-se um sumário comparativo entre os sistemas *Bar Code* e RFID.

Tabela 1.3: Sumário comparativo entre os sistemas *Bar Code* e RFID

Sistema	Bar Code	RFID
Transmissão de dados	Ótico	Eletromagnetismo
Tamanho de Memória	até 100 bytes	até 128 Kbytes
Posição do Leitor	Linha Visual	Sem Linha Visual
Alcance	Alguns metros (Com linha Visual)	Centímetros até Metros (Depende do Sistema)
Segurança	Baixa	Alta
Suscetibilidade ao Ambiente	Alta	Baixa
Anti-Colisões	Não possível	Possível
Preço	\$0.01	\$0.10 – \$1(<i>Passivas</i>)

1.2.4 RFID e Sistemas de Localização em Tempo Real

Os sistemas de localização em tempo real (RTLS) são normalmente associados ao *Global Positioning System* (GPS), porém este tem apenas um funcionamento correto em sistemas *outdoor* (locais abertos), dado que a maioria dos RTLS estão baseados na comunicação via

satélite. No entanto, não apresenta a mesma eficiência em sistemas *indoor* (locais fechados) como, por exemplo, em armazéns de distribuição [32]. Nos sistemas *indoor* que rastreiam objetos ou pessoas, em movimento, são mais utilizados os sistemas RFID. Destaca-se a diferença entre os sistemas RFID instalados em portal fixo e os RTLS, em que, no primeiro as *tags* apenas são lidas quando passam pelo portal e nos RTLS as *tags* são lidas automaticamente e frequentemente, independente da sua localização instantânea [32]. A estrutura básica de sistemas RTLS construídos com base em RFID são:

- **Leitores Fixos:** Em localizações pré-definidas são instalados leitores que esperam a passagem de objetos/pessoas com tags associadas, de modo aos leitores receberem as informações das *tags* e serem enviadas para o sistema de monitorização (*Host*);
- **Tags Móveis:** As *tags* móveis são transportadas com os objetos/pessoas que são rastreadas. Normalmente são utilizadas *tags* ativas, de modo a serem enviadas informações para os leitores, a intervalos de tempos pré-determinados;
- **Sistema de Monitorização:** Recebem as informações dos itens rastreados pelos sensores.

Nos RTLS são empregues diversos sensores para o rastreamento da informação das *tags* localizadas em cada espaço. Estes sistemas poderão tornar-se mais completos, fornecendo informação adicional se, para além da informação da *tag*, o leitor também medir a intensidade do seu sinal, de modo a calcular distância *tag*-leitor. Se no processo envolvido forem utilizados pelo menos três sensores, é possível o cálculo da localização exata da pessoa/objeto pela técnica de triangulação de sinal [32].

1.3 Indústria do Mármore

O termo rocha ornamental tem diversos significados e abrange usualmente as rochas qualificadas de revestimento, podendo ser entendido como todos os materiais provenientes de rochas com funções de estética e estrutura utilizadas em construção civil, que se apresentam como produto final sob a forma de blocos ou placas [43].

Portugal é um dos principais produtores de rochas ornamentais do mundo e a sua extensa variedade de pedra natural enquadra-o numa posição de destaque mundial, com grande poder de geração de riqueza e sustentabilidade do sector [44]. De acordo com [45], a utilização das rochas ornamentais faz parte da tradição Portuguesa e a sua diversidade de utilização vai desde as calçadas até aos monumentos históricos. A mesma referência [45] assegura que: “- *Bastará uma viagem pelo território hoje reconhecido como Portugal, para encontrar a história nacional, inscrita na Pedra e pela Pedra*” .

1.3.1 Impactos Económicos

A atividade de mineração e exploração do subsolo efetuada como conhecemos hoje em Portugal foi iniciada em 1852 com as políticas de modernização do país, através de investimentos

em rodovias e ferrovias, pontes metálicas e novos processos produtivos. O lançamento da Lei de Minas de 1852 foi a abertura de portas que o setor precisava para a receção de investimentos estrangeiros e satisfazer a procura existente de matérias-primas por parte das indústrias de transformação da Europa Central [44]. Apesar de ser uma das atividades mais antigas, em 2012 as rochas ornamentais obtiveram crescimentos e a indústria nacional do sector (extração e transformação), produziram uma receita de 623.4 milhões de euros, dos quais 140.2 milhões devidos a transações comerciais das pedreiras e 483.2 milhões devidos a transações comerciais das empresas transformadoras, sendo o mármore a matéria-prima principal, seguida do granito [44].

Segundo [46], na sua tabela de exportações, as rochas ornamentais representam 46% das exportações, os minérios metálicos 46%, seguindo os minerais industriais com 7% e 1% relativamente a outros. No entanto, em 2019, os subsectores de minerais metálicos movimentaram o volume de 466 milhões de euros, enquanto que os minérios de construção movimentaram 473 milhões de euros, dos quais as rochas ornamentais representavam 468 milhões de euros (Mármore e outras rochas carbonatadas corresponderam a 249 milhões de euros) . De acordo com [44], a indústria de rochas ornamentais é tradicionalmente exportadora, principalmente para a França, China, Arábia Saudita, Espanha, Reino Unido e Alemanha. O crescimento médio das exportações entre os anos de 2005-2013 foi de 6.3% e em 2013, o montante total com esta atividade foi de 372.09 milhões de euros . O setor de Mármore e outras rochas carbonatadas, entre os anos de 2018-2019 cresceu em 10.1% nas exportações, o que corresponde a 195.224 toneladas de pedra e a um volume de 23 milhões de euros. Em 2019 foram exportadas 1.104.602 toneladas de mármore (249 milhões de euros), destacando-se a China com 725.164 toneladas, a Argélia com 21.515 toneladas, a Índia com 19.881 toneladas e os EUA com 15.861 toneladas. As saídas para a União Europeia (UE) totalizam as 200.030 toneladas, principalmente para França com 70.720 toneladas, Espanha com 34.887 toneladas, Reino Unido e Alemanha ambos com 23 mil toneladas [46].

1.3.2 Impactos Ambientais

As extrações de rochas ornamentais provocam prejuízos ambientais e sociais, como os impactos visuais (os mais comuns) e sonoros. Entre os impactos visuais, destaca-se a perda de flora da região no qual a pedreira será instalada, desde a sua construção até à extração de blocos a vegetação é destruída. Os impactos sonoros são muito significativos na vizinhança da pedreira, podendo provocar efeitos colaterais na saúde humana, nomeadamente distúrbios de sono, problemas gástricos e doenças circulatórias [47].

Conforme [48], os cinco principais impactos ambientais da extração são: (1) alteração da topografia, (2) ocupação do solo, (3) degradação das águas superficiais e subterrâneas, (4) poluição do ar, (5) poluição visual. A poluição visual (Fig. 1.8) é consequência dos resíduos restantes dos processos de extração e transformação .



Figura 1.8: Pedreiras em Vila Viçosa, Portugal (Adaptado:Google Earth)

A mineração é dos setores industriais com os maiores índices de geração de resíduos no mundo inteiro. Em 2008, relatou-se que 27.8 % de todos os resíduos gerados na Europa foram de origem da indústria mineira, o que representou mais de 700 milhões de toneladas de resíduos, dos quais 600 milhões de toneladas constituídas por solos e rochas, correspondendo assim 1.5 toneladas per capita [49]. Os resíduos provenientes da extração e transformação das rochas ornamentais representam aproximadamente 83% de perdas da matéria-prima. Em termos médios, para a produção de 330 m^2 de chapas, é necessário extrair um bloco maciço de 30 m^3 , ficando cerca de 20 m^3 nas pedreiras sob a forma de resíduos [50].

2 Digitalização da Indústria do Mármore

Este capítulo tem como objetivo apresentar o conjunto de dados que serão informatizados, provenientes de ensaios não destrutivos realizados no mármore, que auxiliam na tomada de decisão durante o processo de corte dos blocos de pedra [25]. Os dados gerados em cada ensaio são objecto de investigação do projeto científico BRO-CQ [3], são armazenados em bases de dados e localizados através de informação associada a *tags*, identificadoras de cada bloco de mármore. Este processo de incrementar a digitalização na indústria do mármore é o objecto central do presente trabalho, especificamente detalhado no próximo capítulo 3.

2.1 Ensaios Electro Resistivos

O objetivo dos levantamentos elétricos é determinar a distribuição da resistividade do subsolo através de medições à sua superfície. A partir dessas medições, a verdadeira resistividade da subsuperfície pode ser estimada. A resistividade do solo está relacionada com vários parâmetros geológicos, nomeadamente o conteúdo mineral e fluido, porosidade e grau de saturação de água na rocha. Os levantamentos de resistividade elétrica têm sido usados por muitas décadas em investigações hidrogeológicas, de mineração e geotécnicas [51].

A lei física fundamental usada em pesquisas de resistividade é a Lei de Ohm, que rege o fluxo de corrente no solo. A equação da Lei de Ohm em forma vetorial para o fluxo de corrente num meio contínuo é dada por:

$$J = \sigma \cdot E \quad (2.1)$$

Onde σ está a condutividade do meio, J a densidade da corrente e E a intensidade do campo elétrico. Na prática, o que se mede é o potencial de campo elétrico. Notamos que em levantamentos geofísicos a resistividade média, que é igual à recíproca da condutividade ($\rho = 1 / \sigma$), é mais comumente usada [51]. A relação entre o potencial elétrico e a intensidade do campo é dada por:

$$E = -\nabla \cdot \phi \quad (2.2)$$

combinando as equações 2.1 e 2.2, obtem-se:

$$J = -\sigma \cdot \nabla \cdot \phi \quad (2.3)$$

Em quase todas as pesquisas, as fontes atuais estão na forma de fontes pontuais.

Neste caso, sobre um volume elementar ΔV em torno da fonte de corrente I , localizado em (x_s, y_s, z_s) a relação entre a densidade de corrente e a corrente é dada por:

$$\nabla J = \left(\frac{I}{\Delta V}\right) \cdot \delta(x - x_s) \cdot \delta(y - y_s) \cdot \delta(z - z_s) \quad (2.4)$$

Esta é a equação básica que fornece a distribuição de potencial no solo devido a uma fonte de corrente pontual. Um grande número de técnicas foi desenvolvido para resolver esta equação. Estamos perante um problema de modelação "direta", ou seja, pretende determinar-se o potencial que seria observado ao longo de uma determinada estrutura de superfície. Métodos totalmente analíticos têm sido usados para casos simples, como uma esfera num meio homogêneo ou uma falha vertical entre duas áreas, cada uma com resistividade constante [51, 52].

Considerando o caso mais simples de uma subsuperfície homogênea e uma fonte de corrente de ponto único, na superfície do solo, nesse caso, a corrente flui radialmente para longe da fonte e o potencial varia inversamente com a distância à fonte de corrente. As superfícies equipotenciais têm uma forma hemisférica e o fluxo de corrente é perpendicular à superfície equipotencial [52, 53], conforme ilustrado na figura 2.1.

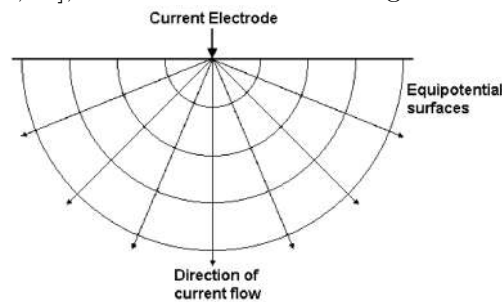


Figura 2.1: Ondas de Campo (Source: [51])

Neste caso, o potencial eléctrico é dado por:

$$\phi = \frac{\rho I}{2Hr} \quad (2.5)$$

Onde r é a distância de um ponto do meio (incluindo a superfície do solo) ao eléctrodo. Na prática, todas as pesquisas de resistividade usam pelo menos dois eléctrodos de corrente, uma corrente positiva e uma fonte de corrente negativa. Os valores de potencial têm um padrão simétrico em torno da posição vertical, no ponto médio entre os dois eléctrodos. Neste caso, o valor de potencial eléctrico do meio é dado por:

$$\phi = \frac{\rho I}{2Hr} \left(\frac{1}{r_{c1}} - \frac{1}{r_{c2}} \right) \quad (2.6)$$

Onde r_{c1} e r_{c2} são distâncias do ponto a partir do primeiro e do segundo eléctrodos de corrente. Normalmente, na maioria dos levantamentos, é medida a diferença de potencial

entre dois pontos (normalmente na superfície do solo). Um arranjo típico com 4 elétrodos é mostrado na figura 2.2.

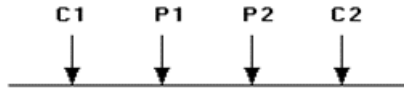


Figura 2.2: Configuração de 4 Elétrodos (Source: [51])

A diferença de potencial é, neste caso, dada por:

$$\Delta\phi = \frac{\rho I}{2Hr} \left(\frac{1}{r_{C1P1}} - \frac{1}{r_{C2P1}} - \frac{1}{r_{C1P2}} + \frac{1}{r_{C2P2}} \right) \quad (2.7)$$

A equação acima define o potencial que seria medido num meio homogêneo com uma matriz de 4 elétrodos. Pesquisas de campo reais são invariavelmente conduzidas em meios não homogêneos onde a resistividade sub-superficial tem uma distribuição 3-D. As medições de resistividade ainda são feitas injetando corrente no solo através dos dois elétrodos de corrente (C1 e C2) e medindo a diferença de tensão resultante nos dois elétrodos de potencial (P1 e P2). A partir dos valores de corrente (I) e potencial ($\Delta\phi$), é calculado o valor de resistividade aparente (ρ_a), conforme eq. 2.8.

$$\rho_a = k \left(\frac{\Delta\phi}{I} \right) \quad (2.8)$$

$$k = \frac{2\pi}{\left(\frac{1}{r_{C1P1}} - \frac{1}{r_{C2P1}} - \frac{1}{r_{C1P2}} + \frac{1}{r_{C2P2}} \right)}$$

Em que k é a constante referente ao arranjo dos instrumentos de medição, que normalmente fornecem um valor de resistência, $R = \frac{\Delta\phi}{I}$. Então, na prática, é calculado o valor de resistividade aparente através da eq. 2.9:

$$\sigma = k \cdot R \quad (2.9)$$

O valor calculado de resistividade não representa a verdadeira resistividade da subsuperfície, mas um valor "aparente" que é a resistividade de um aterramento homogêneo que apresenta um mesmo valor de resistência para o mesmo arranjo de elétrodos. A relação entre a resistividade "aparente" e a resistividade "verdadeira" é uma relação complexa. Determinar a verdadeira resistividade de subsuperfície, a partir dos valores de resistividade aparente, é designado como o problema de "inversão" [53].

A Figura 2.3 mostra os arranjos comuns usados em pesquisas de resistividade, juntamente com seus fatores geométricos. Existem mais dois métodos elétricos que estão intimamente relacionados com o método de resistividade: o método de Polarização Induzida (IP) e o método da Polarização Espectral Induzida (SIP) - também conhecido como método

de Complex Resistivity (CR) [51, 54]. Ambos os métodos requerem instrumentos de medição que são mais sensíveis do que o método de resistividade normal, também com correntes eléctricas significativamente mais altas. Pesquisas de IP são comparativamente mais comuns, particularmente em pesquisas de exploração mineira. A Pesquisa de IP é capaz de detectar minerais condutores de concentrações muito baixas que, de outra forma, poderiam não ser detectados por pesquisas de resistividade ou EM. Pesquisas comerciais SIP são comparativamente raras, embora seja um tema de investigação corrente. Os levantamentos IP e SIP usam correntes alternadas (no domínio da frequência) de frequências muito mais altas do que os levantamentos de resistividade padrão. O acoplamento eletromagnético é um problema sério em ambos os métodos. Para minimizar o acoplamento eletromagnético, a matriz dipolo-dipolo (ou polo-dipolo) é comumente usada [54].

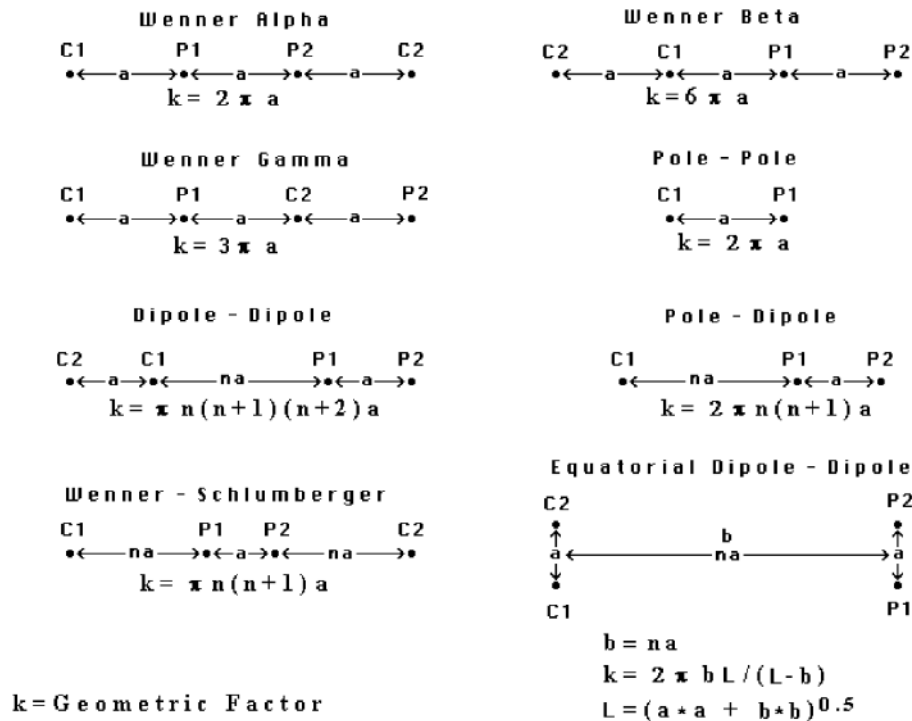


Figura 2.3: Arranjos de Eléttodos (Source: [51])

2.1.1 A relação entre a geologia e a resistividade

A corrente eléctrica flui em materiais terrestres em profundidades rasas por meio de dois métodos principais: condução eletrónica e condução eletrolítica. Na condução eletrónica, o fluxo de corrente ocorre por meio de elétrons livres, como nos metais. Na condução eletrolítica, o fluxo de corrente ocorre por meio do movimento de iões nas águas subterrâneas. Nas pesquisas ambientais e de engenharia conduzidas, a condução eletrolítica é provavelmente o mecanismo mais comum. A condução eletrónica é importante, nas pesquisas mineiras, quando minerais condutores estão presentes, como sulfetos metálicos e grafite [54].

Segundo [55–57], a resistividade das rochas comuns, materiais do solo e produtos

químicos é mostrada na tabela 2.1. Usualmente as rochas ígneas e metamórficas apresentam altos valores de resistividade. A resistividade dessas rochas depende muito do grau de fraturação e da percentagem das fraturas preenchidas com água subterrânea. Assim, um determinado tipo de rocha pode ter uma grande faixa de resistividade, de cerca de 1000 a 10 milhões de Ωm , dependendo se está húmida ou seca. Esta característica é útil na detecção de zonas de fratura e outras características de intemperismo, como em pesquisas de engenharia de águas subterrâneas. As rochas sedimentares, que geralmente são mais porosas e possuem maior teor de água, normalmente apresentam valores de resistividade mais baixos em comparação com as rochas ígneas e metamórficas. Os valores de resistividade variam de 10 a cerca de 10.000 Ωm , com a maioria dos valores abaixo de 1000 Ωm .

Os levantamentos de resistividade fornecem uma imagem da distribuição da resistividade do subsolo. Para converter a imagem de resistividade numa imagem geológica, é importante algum conhecimento dos valores típicos de resistividade para diferentes tipos de materiais de sub-superfície e da geologia da área pesquisada. A resistividade da água subterrânea varia de 10 a 100 Ωm , dependendo da concentração de sais dissolvidos [54]. Observa-se uma baixa resistividade na água do mar (cerca de 0,2 Ωm) devido ao teor de sal relativamente elevado. Esta característica torna este método da resistividade como uma técnica ideal para mapear a interface salina-água doce, em áreas costeiras. Os valores de resistividade nesses casos têm uma faixa muito maior, comparativamente com outras quantidades físicas mapeadas por diferentes métodos geofísicos. A resistividade de rochas e solos, numa área de levantamento, pode variar em várias ordens de grandeza. Em comparação, os valores de densidade usados por levantamentos de gravidade geralmente alteram-se num fator inferior a 2, e as velocidades sísmicas geralmente não se alteram num fator superior a 10 [54]. Todas estas características tornam a resistividade e outros métodos elétricos ou eletromagnéticos, baseados em técnicas geofísicas, muito utilizados e versáteis.

Os valores de resistividade de vários contaminantes industriais metais, como o ferro, têm valores de resistividade extremamente baixos. Produtos químicos que são eletrólitos fortes, como cloreto de potássio e cloreto de sódio, podem reduzir muito a resistividade da água subterrânea para menos de 1 Ωm , mesmo em concentrações razoavelmente baixas [54]. O efeito de eletrólitos fracos, como o ácido acético, é comparativamente menor. Hidrocarbonetos, como o xileno, apresentam normalmente valores de resistividade muito elevados ($6,998 \cdot 10^{16}\Omega\text{m}$). No entanto, na prática, a percentagem de hidrocarbonetos numa rocha ou solo é geralmente muito reduzida pelo que não apresenta um efeito significativo na resistividade na massa [54].

Tabela 2.1: Resistividade dos Materiais

Material	Resistividade ($\Omega.m$)	Condutividade (Siemens/m)
Rochas Metamórficas		
Granito	$5 \cdot 10^3 - 10^6$	$10^{-6} - 2 \cdot 10^{-4}$
Basalto	$10^3 - 10^6$	$10^{-6} - 10^{-3}$
Ardósia	$6 \cdot 10^2 - 4 \cdot 10^7$	$2.5 \cdot 10^{-8} - 1.7 \cdot 10^{-3}$
Mármore	$10^2 - 2.5 \cdot 10^8$	$4 \cdot 10^{-9} - 10^{-2}$
Quartzito	$10^2 - 2 \cdot 10^8$	$5 \cdot 10^{-9} - 10^{-2}$
Rochas Sedimentares		
Arenito	$8 - 4 \cdot 10^3$	$2.5 \cdot 10^{-4} - 0.125$
Xisto	$20 - 2 \cdot 10^3$	$5 \cdot 10^{-4} - 0.05$
Calcário	$50 - 4 \cdot 10^2$	$2.5 \cdot 10^{-3} - 0.02$
Solos e Águas		
Aluvião	10-800	$1.25 \cdot 10^{-3} - 0.1$
Lençóis Freáticos	10-100	$0.01 - 0.1$
Água do Mar	0.2	5
Produtos Químicos		
Ferro	$9.074 \cdot 10^{-8}$	$1.102 \cdot 10^7$
0.01 M Cloreto de Potássio	0.708	1.413
0.01 M Cloreto de Sódio	0.843	1.185
0.01 M Ácido Acético	6.13	0.163
Xileno	$6.998 \cdot 10^{16}$	$1.429 \cdot 10^{-17}$

2.1.2 Processamento dos Dados

No presente trabalho, os ensaios de resistividade utilizaram o arranjo dipolo-dipolo para aquisição dos dados de resistividade aparente, no qual os elétrodos A e B (corrente) e os elétrodos M e N (potencial) fazem a leitura da diferença de potencial no ponto medido. Com os resultados das medições de correntes e diferenças de potencial calcula-se a resistência aparente do bloco.

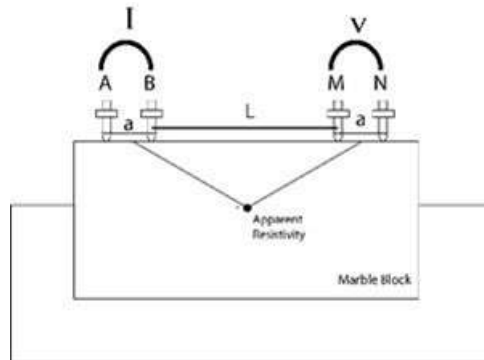


Figura 2.4: Configuração do Arranjo Implementada

Os dados experimentais relativos a: posições dos elétrodos, resistência, diferença de potencial, corrente e resistividade aparente são organizados num ficheiro “.txt” para serem posteriormente processados e permitirem a resolução do problema de “Inversão” através do uso do software RES2DINV/ RES3DINV [58]. Este software resolve o problema da inversão usando o algoritmo dos mínimos-quadrados de Gauss-Newton [53] e por fim gera a imagem do perfil de resistividade em 1D, 2D e 3D.

resistivity_test_output.txt - Notepad

File Edit Format View Help

METHODE = DIPOLE-DIPOLE

a[m] = 0,1

a	b	m	n	V/I [OHMS]	V[V]	I[A]	R[Ohms*m]
1	2	3	4	4.4300E+04	2.0780E+00	4.7000E-05	8.3210E+04
2	3	4	5	8.2100E+04	3.9520E+00	4.8000E-05	1.5430E+05
3	4	5	6	6.1500E+05	2.9850E+00	5.0000E-06	1.1590E+06
4	5	6	7	2.71E+06	2.7111E+00	1.0000E-06	5.1110E+06

Ln 2, Col 10 100% Windows (CRLF) UTF-8

Figura 2.5: Dados Estruturados - Resistividades

Após o processamento dos dados do ficheiro txt, é gerada uma imagem com o perfil das resistividades do bloco de pedra analisado, sendo esta a informação que será adicionada à base de dados gerada no contexto da estratégia de digitalização implementada. Ao inserir a informação na base de dados, esta será associada a uma chave primária e única comum ao bloco de pedra analisado, sendo este assunto apresentado com mais detalhes no capítulo 3.

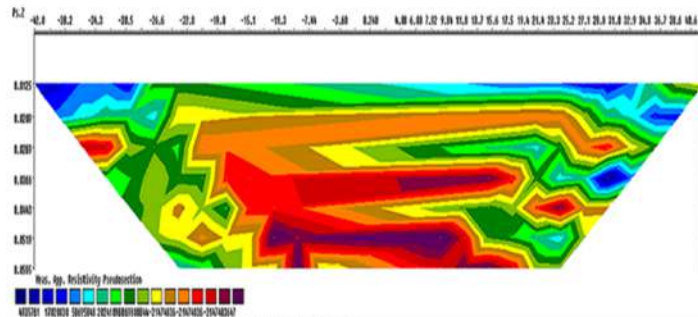


Figura 2.6: Perfil de Resistividades

2.2 Ensaios de Ultrassons

A caracterização de materiais rochosos por métodos ultrassónicos é uma técnica bem conhecida, amplamente utilizada. Desde 1972, medições ultrassónicas de velocidade têm sido aplicadas para determinar a qualidade das pedras naturais, a fim de demonstrar homogeneidade e grau de alteração [59]. Em [60] aplicou medições de velocidade ultrassónica à conservação de pedra e diagnóstico do estado de alteração em esculturas de mármore, introduzindo uma nova metodologia não destrutiva para a conservação da pedra. As técnicas ultrassónicas são possíveis de serem aplicadas tanto no estaleiro, como em laboratório e vários trabalhos de investigação [61–66] estudaram a relação entre as propriedades físicas das rochas e os seus parâmetros ultrassónicos. Duas propriedades interessantes são: (i) a velocidade dos ultrassons está intimamente relacionada com as propriedades dinâmicas das rochas, (ii) existe uma série de fatores que influenciam a atenuação e a velocidade das ondas ultrassónicas nas rochas, nomeadamente, o tamanho, forma e tipo do grão, a densidade, propriedades elásticas, porosidade e anisotropia da rocha [66–68]. No domínio da engenharia de materiais, as técnicas de velocidade do som (VS) têm sido cada vez mais usadas para determinar as propriedades dinâmicas de rochas. A velocidade do som de um maciço rochoso está intimamente relacionada com as propriedades intactas da rocha e a propagação das ondas sonoras nesses maciços rochosos é dependente da estrutura e textura da rocha [66].

O equipamento necessário aos ensaios ultrassônicos consiste num emissor (gerador) a partir do qual os impulsos ultrassônicos são transmitidos; num receptor onde os impulsos são recebidos e um equipamento processador do tempo de viagem do som, através da pedra, desde o transmissor ao receptor (Figura 2.7).

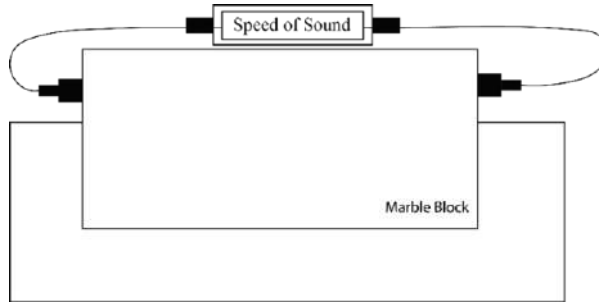


Figura 2.7: Ensaio ultrassônico

Um impulso ultrassônico é criado pela aplicação de uma rápida mudança de potencial de um transmissor-driver a um elemento piezoelétrico que provoca a sua vibração natural (frequência fundamental). O transdutor é colocado em contato com o material para que as vibrações sejam transferidas para o material. As vibrações viajam pelo material e são captadas pelo receptor. A velocidade da onda é calculada usando o tempo gasto pelo impulso para percorrer a distância entre o transmissor e o receptor. Os transdutores são mantidos firmemente nas superfícies das amostras, em contacto com um gel condutor acústico para permitir o bom contato com a amostra [69]. A velocidade de propagação do impulso é dada pela expressão:

$$V = \frac{L}{t} \quad (2.10)$$

onde V é a velocidade [m/s], L o comprimento longitudinal do bloco [m] e t o tempo de viagem [s].

A relação entre velocidade e densidade nas rochas é geralmente considerada linear [62]. O tempo de trânsito das ondas sonoras depende da densidade dos minerais que formam as rochas [70]. A velocidade dos impulsos ultrassônicos num material sólido depende da densidade e das propriedades elásticas desse material. A qualidade de alguns materiais está usualmente relacionada com a sua rigidez elástica, pelo que a medição da velocidade do impulso ultrassônico em tais materiais pode frequentemente ser usada como indicador da qualidade da sua estrutura, bem como para determinar as suas propriedades elásticas [71]. O módulo de elasticidade é uma importante propriedade mecânica das rochas pois caracteriza-as relativamente ao seu potencial uso como materiais de construção. É o parâmetro que determina a deformabilidade do material sob cargas aplicadas, tornando-se uma propriedade essencial para qualquer elemento estrutural [72–74]. Os testes de deformabilidade exigem que as amostras sejam extraídas e as cargas aplicadas em ambiente de laboratório. A natureza

destrutiva deste teste significa que não é adequado para uso em certas situações, como em edifícios históricos. Alternativamente, o módulo de elasticidade pode ser obtido a partir de ensaios não destrutivos: normalmente usando resultados obtidos de ensaios que medem a velocidade de propagação de ondas elásticas ultrassônicas, também chamado de módulo dinâmico [75–79]. O módulo estático (E), obtido a partir de procedimentos mecânicos convencionais de laboratório, é necessário para calcular ou modelar as deformações de um edifício sob carga em serviço. Nos casos em que não é possível determinar as características da rocha por meio de ensaios destrutivos, o uso de técnicas não destrutivas por meio de dispositivos móveis constitui uma alternativa adequada [76].

Em [66], foram realizados testes ultrassônicos em rochas carbonáticas turcas (calcários, mármore, dolostones, calcários dolomíticos) e compararam os valores de velocidade ultrassônica com medições diretas de resistência à compressão uniaxial, módulo de *Young* e densidade. Os resultados experimentais indicam fortes correlações lineares positivas entre as propriedades mecânicas e os valores de velocidade ultrassônica. Tendências semelhantes foram encontradas por [80], que correlacionaram a resistência à compressão, resistência à tração, resistência ao corte, densidade, módulo de *Young*, coeficiente de *Poisson* e velocidade da onda, para amostras de carvão, xisto e arenito de minas indianas.

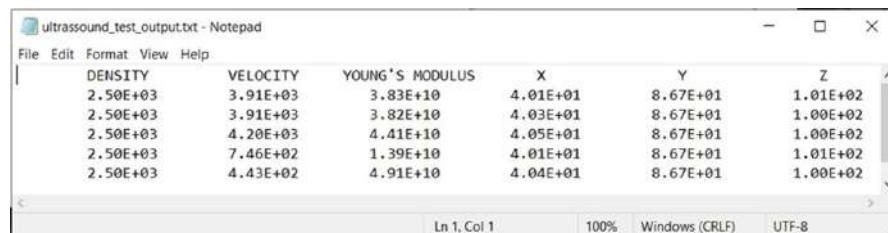
Deste modo, a eq. 2.11 ilustra a relação entre a velocidade do som, o módulo de elasticidade de *Young* e a densidade do bloco de pedra:

$$V = \sqrt{\frac{Y}{\rho}} \quad (2.11)$$

onde v é a velocidade do som, Y o módulo de elasticidade de *Young* e ρ a densidade.

2.2.1 Processamento dos Dados

Considerando que um bloco de mármore apresenta densidade homogênea para cada localização (x, y, z), realizou-se um conjunto de medições de velocidade ultrasônica em todo o bloco. Aplicando a equação (2) aos resultados das medições, foi calculado o módulo de *Young*, conforme ilustrado na Figura 2.8.



DENSITY	VELOCITY	YOUNG'S MODULUS	X	Y	Z
2.50E+03	3.91E+03	3.83E+10	4.01E+01	8.67E+01	1.01E+02
2.50E+03	3.91E+03	3.82E+10	4.03E+01	8.67E+01	1.00E+02
2.50E+03	4.20E+03	4.41E+10	4.05E+01	8.67E+01	1.00E+02
2.50E+03	7.46E+02	1.39E+10	4.01E+01	8.67E+01	1.01E+02
2.50E+03	4.43E+02	4.91E+10	4.04E+01	8.67E+01	1.00E+02

Figura 2.8: Dados Estruturados - Ultrassom

Os dados de densidade, velocidade, módulo de *Young* e localização (x, y, z) são organizados e armazenados num arquivo de texto (.txt) e depois colocados na base de dados do sistema, associados a uma chave única associada à pedra (*tag*). Este processo é apresentado com maior detalhe no capítulo 3.

2.3 Ensaios de Imagem

A visão artificial consiste no processamento de imagens por meio de um processador/computador. Os principais objetivos do processamento de imagens são o reconhecimento e a compreensão automática das imagens [81]. Essas tarefas incluem detecção de cores ou tons, detecção de objetos e fronteiras, segmentação, classificação e identificação de imagens. A tecnologia de processamento de imagem desenvolveu-se rapidamente a partir de 1960/1970 [81, 82]. O sistema de processamento de imagem contém três elementos fundamentais: um elemento de aquisição de imagem, um elemento de processamento de imagem e um elemento de exibição de imagem. A aquisição de imagem é o processo de obtenção de uma imagem digital de um objeto usando uma camera ou scanner [81]. O elemento de processamento de imagem faz a filtragem (aplicação de operadores matemáticos à matriz da imagem), a análise e a codificação das imagens [81, 82]. Os métodos de processamento de imagens são usados transversalmente: na biologia e ciências médicas para avaliação de imagens biomédicas; na automação industrial para reconhecimento de padrões e inspeção visual automática; na aeronáutica e ciências espaciais para avaliação de fotos de satélite e previsão do tempo; na defesa e segurança; etc.

No caso concreto deste trabalho, a área de aplicação do processamento de imagem envolve as ciências da terra e a indústria mineira. As técnicas de processamento de imagens são usadas nesta indústria para controlar os efeitos ambientais das minas [83]; para determinar algumas características de maciços rochosos [84]; na identificação de minerais e estimativa do conteúdo de metal [85, 86]; para determinar a distribuição do tamanho dos agregados [87, 88]; para controlar remotamente os circuitos de britagem e moagem [89]; para controle do tamanho da espuma na flotação [90–92][40–42]; entre outros.

No caso da produção de mármore para construção e pavimentação, a classificação do mármore de acordo com a sua finalidade e uso, é um procedimento importante, em que a classificação dos produtos de mármore é realizada por um operador humano, observando as propriedades físicas, fundamentalmente a estrutura e a cor. As propriedades físicas do mármore, como dureza, tamanho de grão, fratura, estrias, porosidade, tamanho, orientação, espessura, distribuição de juntas preenchidas, brilho e resistência são usadas como critérios de classificação, pelos especialistas geólogos [93–97] .

O presente trabalho pretende mostrar uma possibilidade para a aplicação da inspeção visual automática da superfície dos blocos de pedra, de modo a poder-se digitalizar este processo, com o correspondente armazenamento da informação em base de dados. Naturalmente que este não é o primeiro trabalho de classificação de imagens de mármore. Numerosos trabalhos científicos publicados mostram modelos de classificação automática de mármore

para facilitar a seleção e classificação de diferentes categorias de mármore em diferentes áreas de produção [94, 98–100]. A maioria dos trabalhos utiliza a textura ou a cor como critérios de classificação. O conteúdo de minerais de mármore produzidos em diferentes áreas geográficas, apresentam diferentes características de textura, fundamentalmente devido ao ambiente geográfico, nomeadamente a temperatura, condições climáticas e pressão atmosférica [100].

Os produtos de mármore têm sua principal aplicação no revestimento de grandes superfícies para decoração, sendo utilizado sob a forma de placas. Devido à grande importância da aparência visual, os requisitos necessários para a qualidade dessas placas não se referem apenas a parâmetros tecnológicos (como resistência ou taxa de polimento), mas também às aparências estéticas, como homogeneidade de cor, textura ou manchas, de modo a garantir um certo nível de uniformidade em toda a superfície. Sendo o mármore um material natural, é altamente heterogêneo não existindo duas placas de mármore completamente idênticas [101]. Estes parâmetros visuais são importantes não apenas nos estágios iniciais de construção, mas também quando melhorias são introduzidas nessas superfícies. Em função desses requisitos torna-se necessário classificar as placas de mármore em classes ou grupos homogêneos, de acordo com sua qualidade e aparência. No entanto, o controle dessas características não é economicamente viável de ser realizado durante a extração da matéria-prima nas pedreiras, pelo que deverá ser realizado no final da linha de produção, onde especialistas humanos avaliam e classificam o produto de acordo com os parâmetros visuais mencionados anteriormente. No entanto, este método de classificação baseado na avaliação visual humana apresenta três problemas principais: i) subjectividade do critério (operadores diferentes podem apresentar critérios diferentes); ii) fadiga visual após um período de tempo, que degrada o desempenho da classificação; iii) condições de luminosidade que podem mudar durante o dia, oferecendo diferentes percepções visuais de um mesmo padrão em diferentes momentos. Todos esses problemas geram resultados altamente dependentes de condições subjetivas e mutáveis, pelo que existe potencial de melhorar as escolhas, com a introdução de critérios objetivos e sistemáticos [101]. A cor é um parâmetro físico principal que influencia as escolhas dos clientes na indústria de mármore. A maioria das empresas produtoras de mármore resumem as propriedades físicas e químicas dos seus produtos em catálogos de produtos. Nestes catálogos normalmente não existe um valor numérico para descrever as cores dos mármore produzidos, dada a sua variabilidade, sendo dada preferência à impressão de catálogos a cores [100].

Um outro parâmetro a ser observado na indústria das rochas ornamentais em geral e que merece destaque, são as descontinuidades presentes nas rochas. As descontinuidades na massa rochosa incluem falhas, juntas, planos de estratificação e outros tipos de fraturas. Com exceção das propriedades de atrito em pequena escala, as propriedades das descontinuidades devem ser determinadas no campo, a partir de furos, escavações e afloramentos naturais. As propriedades de descontinuidade, a serem determinadas no campo, incluem orientação, comprimento, espaçamento, rugosidade, persistência, abertura, enchimento, terminação e outras propriedades [102]. É amplamente aceite que, caracterizar as descontinuidades

é um dos dados mais importantes para o comportamento mecânico e hidráulico de uma massa rochosa [103,104]. O processamento de imagens digitais oferece a possibilidade de identificar fraturas e informações associadas de maneira semiautomática ou automática. Os procedimentos automatizados podem reduzir os erros associados à recolha dos dados de fratura no campo, eliminando assim o viés humano e padronizando o procedimento de amostragem. Este procedimento automático também pode ser utilizado para aumentar a quantidade de informações de fratura que são reunidas rotineiramente num levantamento de campo. Para a indústria mineira, os grandes bancos de dados de fratura podem ser usados para otimizar as várias etapas de quebra de rochas, de processamento dispendioso, nomeadamente no consumo de energia, na detonação, britagem e moagem. Na indústria geotécnica, os bancos de dados de fratura permitem que projetos de engenharia probabilísticos sejam realizados regularmente onde tal análise detalhada não era economicamente viável anteriormente. Além disso, ao criar imagens de faces de rochas à distância, os problemas de segurança e de acesso são reduzidos ou eliminados.

Nos últimos anos, muita investigação tem sido focada na extração de traços de descontinuidade de modelos de superfície 3D [105,106], ou seja, nuvens de pontos 3D de alta resolução de superfícies de maciços rochosos. A técnica de fotogrametria é capaz de obter nuvens de pontos 3D a partir de pares de imagens 2D. A tecnologia *LiDAR* [107] é uma outra solução para a obtenção de nuvens de pontos, também em 3D. Os prós e contras de ambas as técnicas são discutidos por muitos autores [105,108]. Os dois principais métodos podem ser usados para detectar traços usando uma modelização de superfície 3D. Os traços de descontinuidade podem ser obtidos como linhas de intersecção entre os planos de ajuste das superfícies dos maciços rochosos [106,107,109].

2.3.1 Processamento dos Dados

A partir de imagens das superfícies da pedra e aplicando filtros apropriados, utilizando o software MATLAB [110], é possível identificar padrões de veios que podem ser identificados como possíveis linhas de fratura do bloco, conforme ilustrado na figura 2.9.

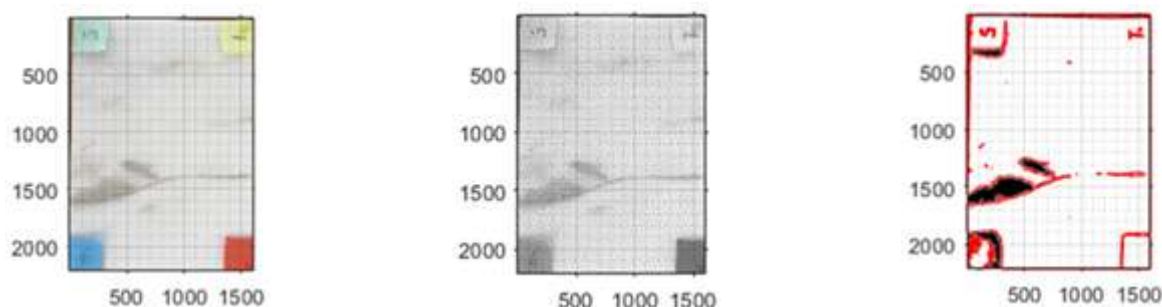


Figura 2.9: Imagem da Superfície - Binarização

Após o tratamento da imagem e identificação dos veios de que podem ser fraturas, é gerada uma imagem tridimensional e um arquivo texto (.txt) com as coordenadas dos veios (figuras 2.10 e 2.11). Por fim, os resultados obtidos são colocados na base de dados do sistema, associados a uma chave única associada à pedra (*tag*). Este processo é apresentado com maior detalhe no capítulo 3.

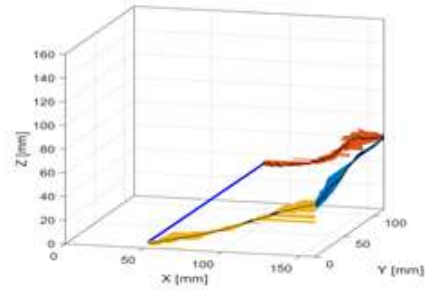


Figura 2.10: Perfil de Descontinuidades

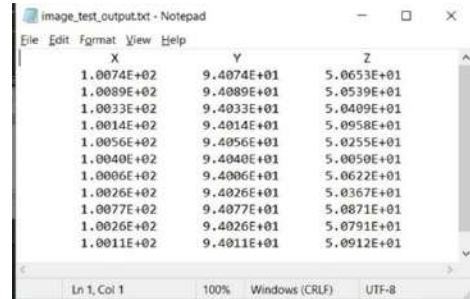


Figura 2.11: Dados Estruturados – Imagem

3 Concepção do Protótipo

Este capítulo tem como objetivo apresentar a estratégia desenvolvida para a concepção de um sistema integrado para a digitalização da indústria do mármore, com identificação automática. São apresentados: i) a estrutura desenvolvida; ii) os equipamentos seleccionados; iii) as comunicações e as lógicas implementadas. A figura 3.1 apresenta o ciclo de vida do mármore, com início no corte dos blocos de pedra e o seu término no consumidor final, passando pelos processos intermédios, nomeadamente a fase de ensaios laboratoriais, cujos testes foram já apresentados no anterior capítulo 2.

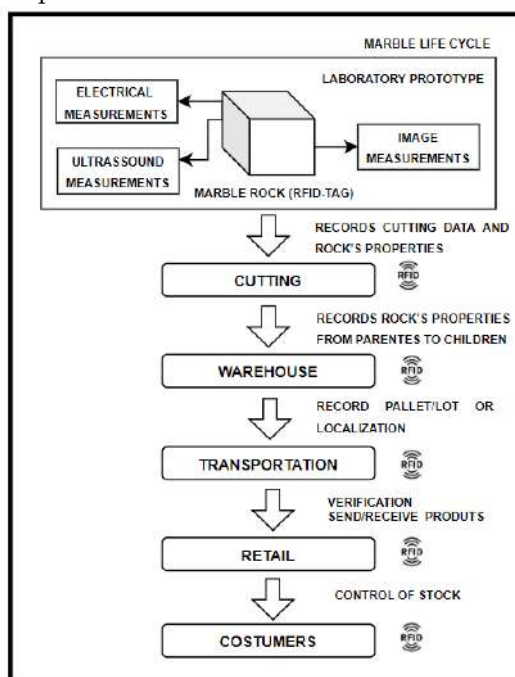


Figura 3.1: Ciclo de Vida do Mármore

O novo sistema automático de integração desenvolvido tem como objetivo o aumento de eficiência da cadeia de valor do mármore, dado que os actuais processos manuais não são suficientemente rigorosos e fiáveis. A tecnologia de integração seleccionada é a identificação por radio frequência (RFID), o que permite o rastreamento de todo o ciclo de vida do produto, que é de extrema importância para os ensaios laboratoriais, dados os resultados correspondentes poderem ser gravados em base de dados online, e acedidos através da identificação exclusiva da sua etiqueta (RFID), agregada fisicamente ao bloco de pedra. Ao permitir a rastreabilidade das pedras, os processos de armazenamento, transporte e venda aos consumidores finais tornam-se mais eficientes, nomeadamente facilitando a localização dos produtos em armazém, reduzindo o tempo de verificações durante o envio/recepção de produtos, facilitando o controlo de *stocks*.

Na figura 3.2 ilustra-se a estrutura do protótipo desenvolvido, no âmbito dos ensaios

laboratorias (ver fig. 3.1) para a digitalização da indústria do mármore, permitindo o registo dos resultados dos ensaios, em base de dados *online* e o seu acesso à área produtiva para optimização dos cortes a executar nos blocos de pedra. Os dois primeiros passos do ciclo de vida do mármore, conforme ilustrado na fig. 3.1, - Ensaios Laboratoriais e Corte - foram objecto de protótipos experimentais, de modo a validar o novo conceito desenvolvido para a digitalização da indústria do mármore, atendendo ao requisito de escalabilidade para a sua aplicação nas restantes fases subseqüentes do ciclo de vida.

A primeira fase – Ensaios Laboratoriais - contempla os testes não destrutivos para identificação da estrutura interna dos blocos de pedra (ensaios Eléctricos, Ultrassónicos e Inspeção Visual Automática), na qual são modeladas as propriedades estruturais, essenciais para a tomada de decisão de cortes optimizados do bloco de pedra (segunda fase – Corte).

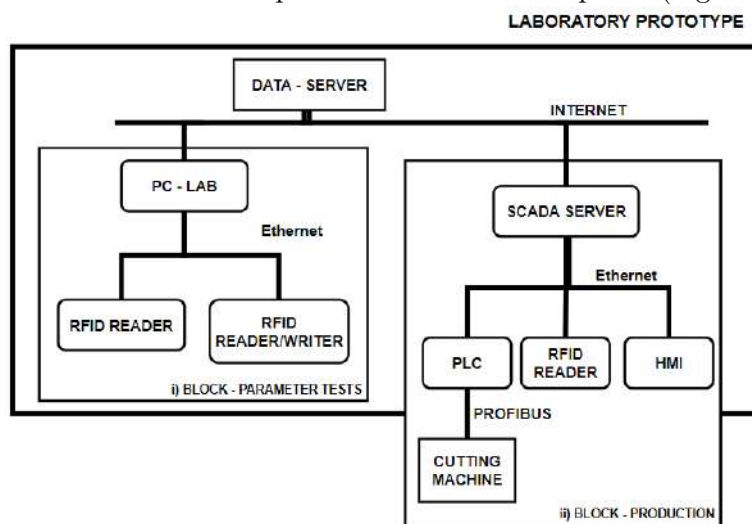


Figura 3.2: Estrutura do Protótipo desenvolvido na Universidade de Évora

Para a fase – Ensaios Laboratoriais – o protótipo desenvolvido utiliza dois computadores, um será responsável pela realização das medições dos ensaios no Bloco de pedra e o outro será conectado aos sensores RFID (identificação do bloco de pedra ensaiado) e realizará a conexão com a base de dados *online* para envio dos dados de localização e armazenamento dos resultados dos ensaios laboratoriais na plataforma Web. Para a fase – Corte - o protótipo desenvolvido utiliza: i) um sensor RFID para detecção da pedra que está a ser analisada; ii) um PLC para o processamento dos dados do RFID e para o envio dos dados do corte optimizado para as máquinas na área produtiva (chão de fábrica); iii) um Painel de Utilizador HMI (*Human - Machine Interface*) para visualização dos dados lidos no PLC; iv) um sistema SCADA para supervisão e controlo de todo o Sistema integrado desenvolvido e para o envio de dados para a base de dados *online*.

O protótipo experimental desenvolvido para a fase - Ensaios Laboratoriais – é ilustrado nas figuras 3.3 e 3.4 .

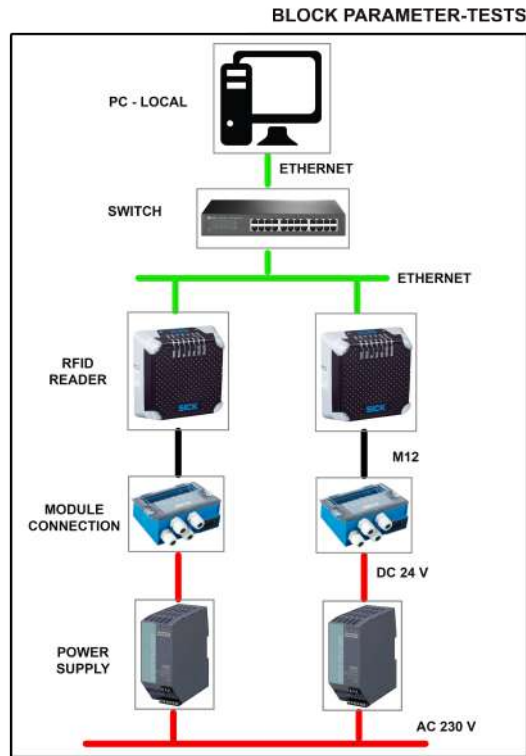
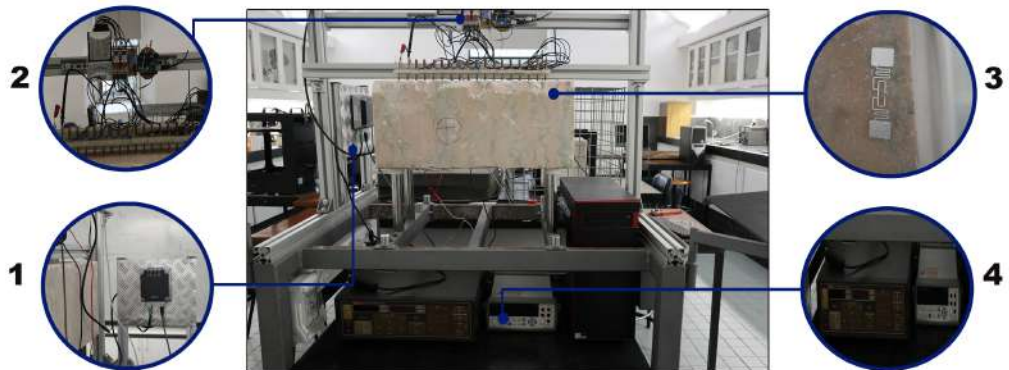


Figura 3.3: Ensaio Laboratoriais (esquema de comunicações)



1. LEITOR RFID (SICK RFU620-10100)
2. SISTEMA DE AQUISIÇÃO National Instruments DAQ-6009
3. TAG PASSIVA (Alien H3 EPC Global Gen 2)
4. FONTE DE CORRENTE Kithley 228A E MULTÍMETRO Agillient 34410A

Figura 3.4: Ensaio Laboratoriais (protótipo implementado na Universidade de Évora)

O protótipo experimental desenvolvido para a fase - Corte – é ilustrado nas figuras 3.5 e 3.6 .

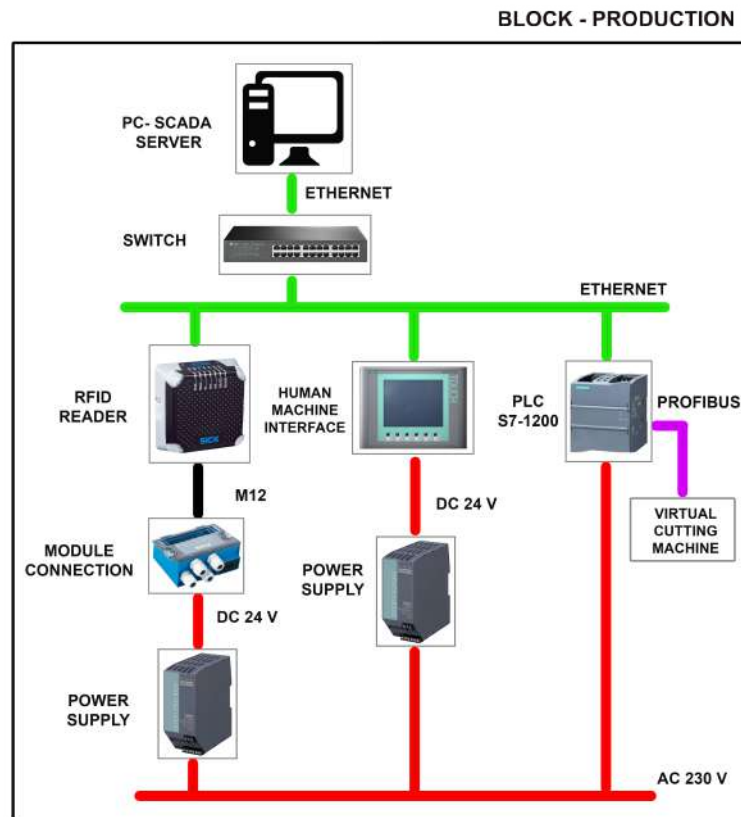
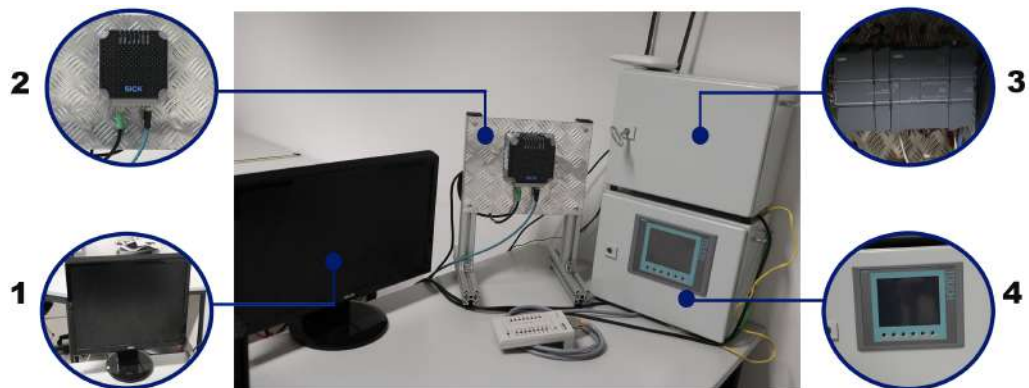


Figura 3.5: Protótipo de Produção Industrial (esquema de comunicações)



1. SERVIDOR SACADA SIEMENS WINCC
2. LEITOR RFID (SICK RFU620-10100)
3. PLC SIEMENS S7-1200
4. HMI SIEMENS KTP600

Figura 3.6: Protótipo de Produção Industrial (implementação na Universidade de Évora)

Nos protótipos experimentais desenvolvidos, o volume de dados que necessitam armazenamento (resultados dos ensaios e identificador do bloco de pedra), excedem em muito as capacidades de armazenamento numa simples etiqueta (*tag*), pelo que foi desenvolvida uma solução que compreende a criação de um protocolo utilizando o identificador único universal da *tag* (UUI - *Universal Unique Identifier*). A solução implementada está esquematizada na figura 3.7, onde o sensor RFID envia para o dispositivo conectado uma string com o UUI e o protocolo de início e de fim. O receptor da string tem um algoritmo desenvolvido para detectar as partes do protocolo, extraíndo o UUI da string, e por fim o dispositivo vai enviar

o UUI para a base de dados online, adicionando dados, nomeadamente: data, hora e área física da operação.

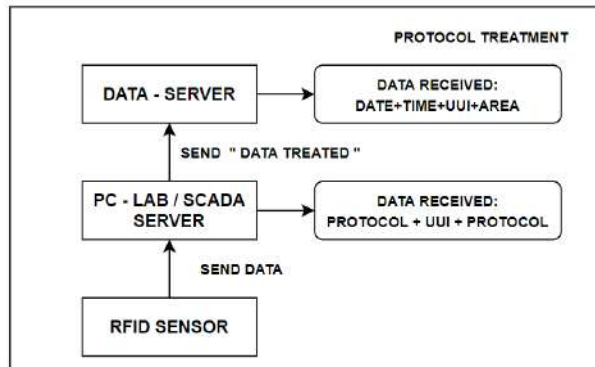


Figura 3.7: Fluxograma do Protocolo de Comunicação

Uma aplicação em linguagem Delphi foi projetada para funcionar com leitores RFID na plataforma PC, seguindo o protocolo desenvolvido. A figura 3.8 mostra a estrutura lógica da aplicação, que tem várias funções desenvolvidas, tais como: ligações TCP/IP; ligação de base de dados online; registos de novos objetos, selecção de áreas físicas.

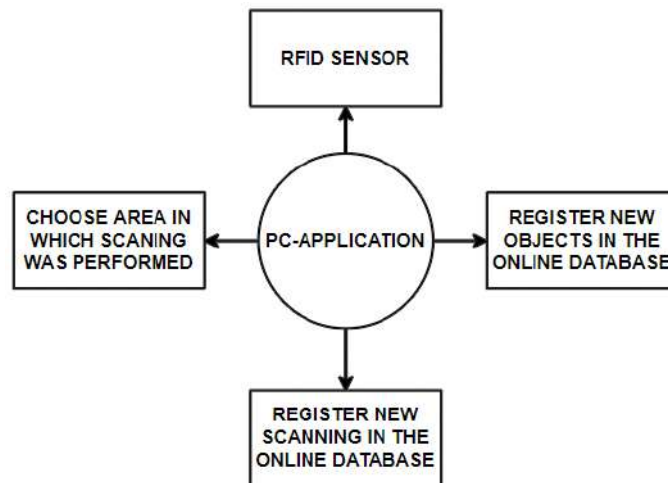


Figura 3.8: Representação Gráfica da Aplicação do PC-APPLICATION

Abaixo é apresentada uma breve explicação das características implementadas:

- **TCP/IP:** A utilização do protocolo foi, em parte, condicionada pelos sensores utilizados, principalmente porque é um protocolo actual, amplamente utilizado e fácil de interagir com os sistemas existentes;
- **BASE DE DADOS ONLINE:** A base de dados online é uma ferramenta necessária quando se pretende desenvolver um sistema em tempo real, fundamentalmente quando comparada com alternativas de armazenar dados, nomeadamente arquivos de texto, Excel ou mesmo bases locais;
- **REGISTAR NOVOS OBJECTOS:** Permite que o sistema operado pelo PC-LAB adicione instantaneamente novos objetos, sempre que necessário. Em sequência, poderão ser anexados os ficheiros de dados;

- **ESCOLHER AREA:** Dado que o sistema tem a capacidade de rastrear os objetos em várias áreas físicas, o utilizador pode indicar a localização do sensor que adquiriu os dados, nomeadamente estação PC-LAB, Instalação do ciclo de vida (transporte, retalho, etc).

Diante das vantagens de se empregar uma base de dados online, desenvolveu-se uma plataforma, também online para visualização e manipulação de dados na base. Com base na figura 3.9, a plataforma pode executar as seguintes tarefas: i) receber dados de aplicações (DATA-HORA-UUI-AREA); ii) registar novos objetos a serem monitorizados; iii) registar novas áreas; iv) registar anexos de arquivos; v) Realizar a busca de objetos.

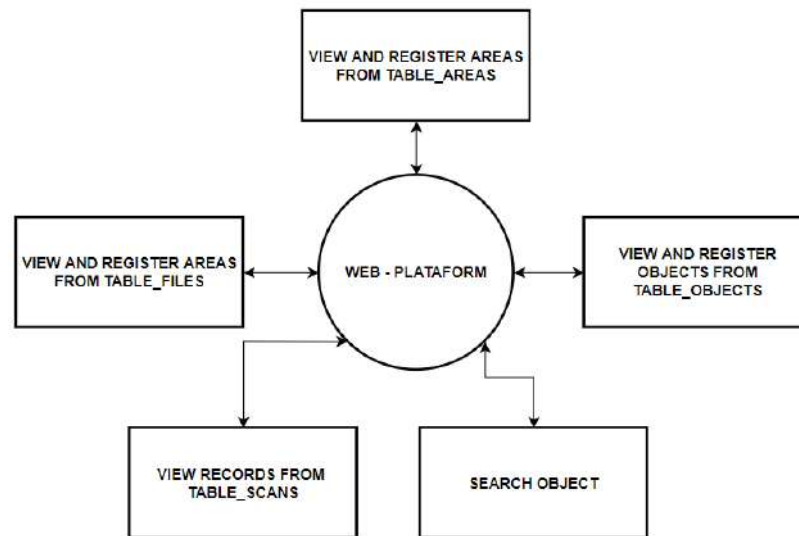


Figura 3.9: Representação Gráfica da Plataforma Web

As seguintes funcionalidades foram implementadas na plataforma web:

- **OBJECTOS:** Regista ou remove novos objetos na tabela de bases de dados ("TABLE_OBJECTS");
- **AREAS:** Regista ou remove novas áreas na tabela de bases de dados ("TABLE_AREAS");
- **FICHEIROS:** Regista ou remove novos ficheiros na tabela de bases de dados ("TABLE_FILES");
- **BUSCAR OBJECTOS:** Consulta a base de dados para registos (digitalizações e ficheiros) a partir do UUI do objeto armazenado.
- **DIGITALIZAÇÕES:** Permite visualizar todas as digitalizações registadas na Base de dados.

Conforme ilustrado na figura 3.10, a base de dados online está organizada em 4 tabelas: *AREAS*; *SCANS*; *OBJECTS*; *FILES*.

Ao realizar-se a digitalização de um objeto, os dados DATA-HORA-UUI-AREA, são adicionados na tabela de SCANS. Porém, se o objeto em questão não estiver registado na base de dados, ele não será visível na plataforma online (em visualizações de scans ou em pesquisa de objetos). A tabela de SCANS possui colunas, que direta ou indiretamente, vai

procurar os respectivos dados noutras tabelas, à medida que o objeto é apontado pelo seu UUI. Ao registá-lo na tabela de objetos, ao UUI é atribuído um código, que será o mesmo código, que será utilizado para o seu registo na base de dados. O mesmo acontece com a tabela AREAS, que quando inserida na base dados é atribuído um código, que estará na tabela de SCANS. Desta forma, a separação dos dados em tabelas, permite uma melhor organização da base de dados, dado que o registo, a consulta e a eliminação de dados de uma tabela torna-se independente do seu registo noutras tabelas. As 4 tabelas interrelacionam-se mas funcionam mais eficientemente.

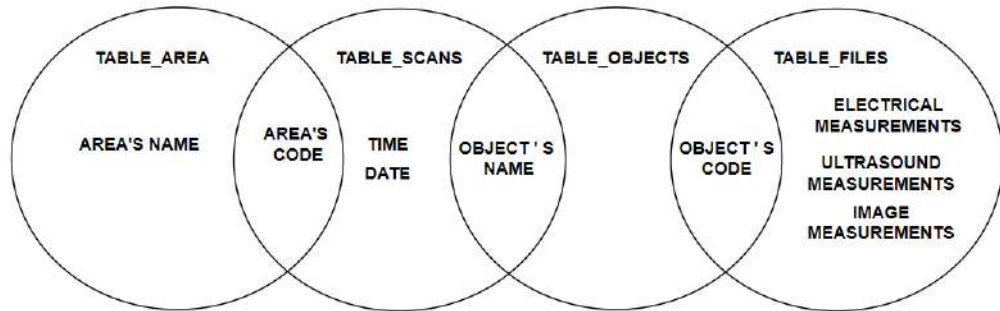


Figura 3.10: Organização da Base de Dados *Online*

Relativamente ao envio de dados do sistema SCADA para a base de dados online, foi necessária a implementação de uma aplicação intermediária, dado que o SCADA apenas se conecta com a base de dados SQL SERVER e a base de dados online foi desenvolvida na plataforma com MYSQL.

Neste contexto, para o sistema realizar o envio dos dados do SCADA, além da base de dados online, tem que se criar uma base de dados local, no sistema SCADA. A aplicação denominada "TRANSFERDATA" tem implementada no seu algoritmo a tarefa de se conectar com ambas as bases de dados e verificar novas entradas na tabela do SQL SERVER. Caso seja detetada uma nova entrada, o algoritmo consulta a tabela em MYSQL para saber se o último objeto adicionado é igual ao que se está a inserir. Se forem iguais, o Objeto é rejeitado, caso seja diferente, o objecto é inserido. A verificação do último dado da tabela tem a finalidade de evitar entradas duplas pelo mesmo dispositivo, visto que os sensores RFID por vezes podem realizar várias leituras do mesmo objeto, num curto espaço de tempo, em função do tempo de permanência do Objeto no seu campo de operação.

Em síntese, foi desenvolvido um *script* em *Visual Basic Script* (VBscript) para enviar os dados do sistema SCADA para o SQL SERVER, com a seguinte sequência: i) O sensor RFID lê e envia os dados para o PLC; ii) o PLC envia os dados lidos para o HMI e para o servidor SCADA; iii) Tanto o HMI como o SCADA mostram ao Utilizador os dados recebidos; iv) Ao atualizar a variável no sistema SCADA é ativado o script VBscript que envia os dados para o SQL SERVER; v) A aplicação "DATATRANSFER", analisa os dados armazenados e toma as decisões de inserção ou rejeição de objetos, de acordo com a regra acima enunciada.

4 Estratégia para Digitalização da Indústria do Mármore

4.1 Configuração dos Leitores RFID

Os leitores RFID utilizados foram o RFU620-10100 da *SICK* [111], que realizam leitura/escrita em *tags* UHF, trabalham na frequência UHF (860 MHz -960 MHz), possuem antena integrada, com um alcance de até 2 m de distância, ligação *ethernet/profinet* para comunicação de dados.

A figura 4.1 mostra o ecrã inicial do software SOPAS da *SICK* [112], no qual é possível criar projetos com vários leitores, sendo destacado que o *software* já detetou os leitores e seus respectivos endereços de *IP*.

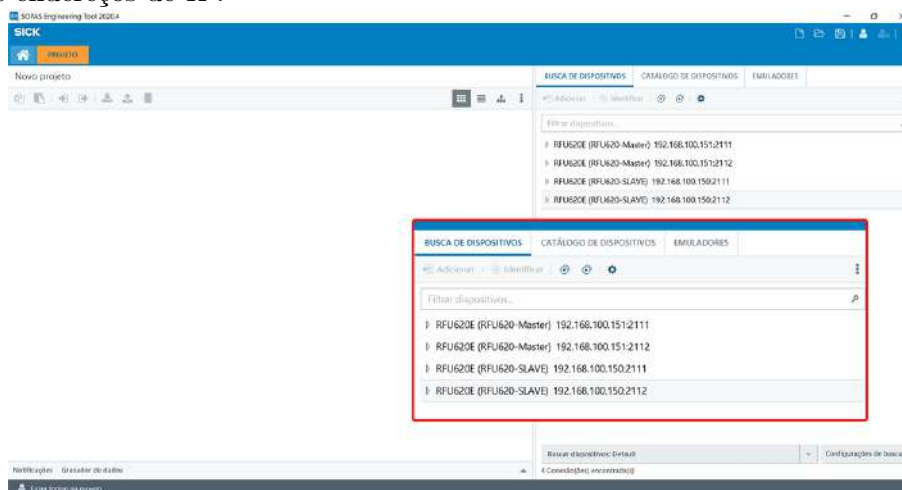


Figura 4.1: SOPAS - Tela inicial

4.1.1 Inicialização Rápida

Ao adicionar leitores ao projeto criado e abrir o separador de definições, existe uma opção “QUICKSTART” (figura 4.2), que serve para realizar testes de verificação das etiquetas (ex.: *tags* que estão na gama de antenas). A visualização dos *Universal Unique Identifier* (UUI) das *tags* pode estar em caracteres ASCII ou *hex*. Ao clicar duas vezes no rato em cima do UUI, abre automaticamente o ecrã de leitura/escrita da etiqueta (figura 4.3).

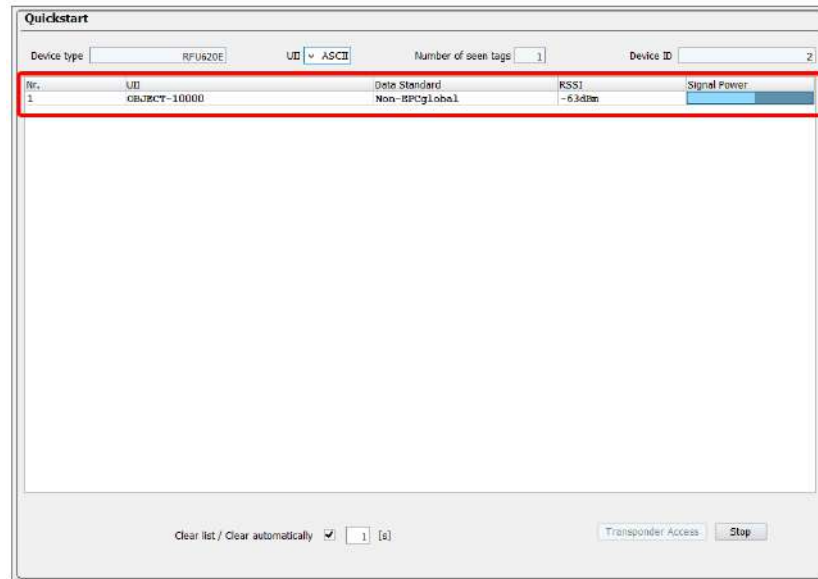


Figura 4.2: SOPAS - Leitura rápida

4.1.2 Leitura/Escrita em Tags

O separador de leitura/escrita permite ver informações de etiquetas, tais como: fabricante, o tamanho da memória interna em bits, o tamanho da memória do UUI em bits, e o que está escrito dentro destas memórias. Realçado na figura 4.3 pode-se ver e modificar o UUI da etiqueta de leitura, em caracteres ASCII, *hex*, binário e VDA.

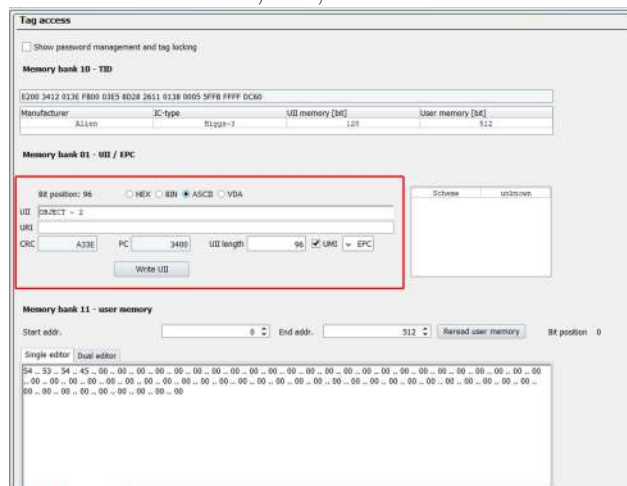


Figura 4.3: SOPAS - Leitura/escrita de dados das Tags

4.1.3 Configurações da Antena

O Leitor RFU620-10100 [111] possui uma antena integrada, e neste caso, tudo é incorporado na unidade fornecida pelo fabricante. No entanto, existe a possibilidade de configurar a sua antena de acordo com a potência emitida (figura 4.4). A potência pode ser ajustada, por exemplo, para efetuar leituras ou escritas, permitindo que o leitor que tem um alcance até 2 m, apenas execute na aplicação utilizada, operações a 50 cm, se necessário.

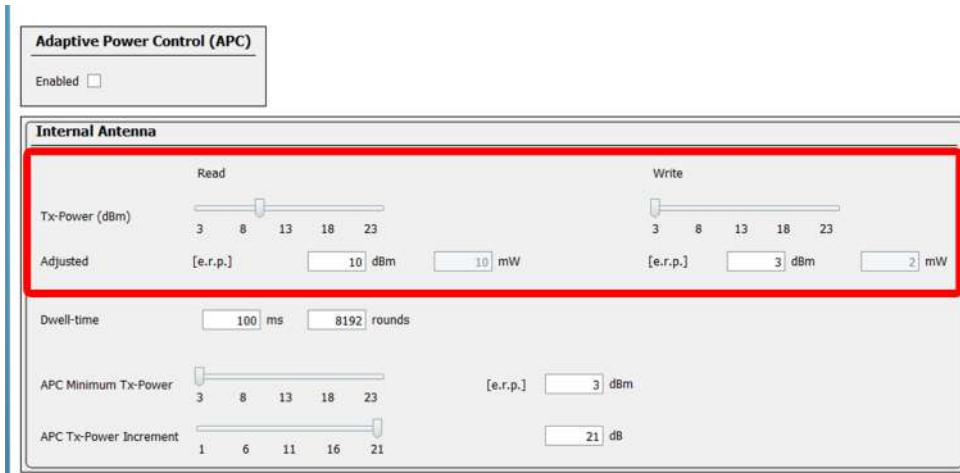


Figura 4.4: SOPAS - Ajuste de alcance da antena

4.1.4 Configurações dos *Triggers* de Leituras

As leituras dos leitores precisam de um *trigger* para serem inicializadas e, a depender do modo de funcionamento, precisam de outro *trigger* para terminar as leituras. Os modos podem basicamente usar *triggers* por entradas externas, por rede CAN, por outros leitores RFID em rede, por caracteres enviados via *ethernet* e por tempo. No trabalho implementado, o modo foi selecionado por tempo (figura 4.5), porque simplifica os códigos que executam a aquisição de dados do leitor, uma vez que nada é enviado para iniciar/terminar leituras e não são esperadas outras entradas. O leitor RFID executa ciclos de leitura sempre que energizado, permitindo o controle do tamanho do ciclo pelos seus temporizadores internos.

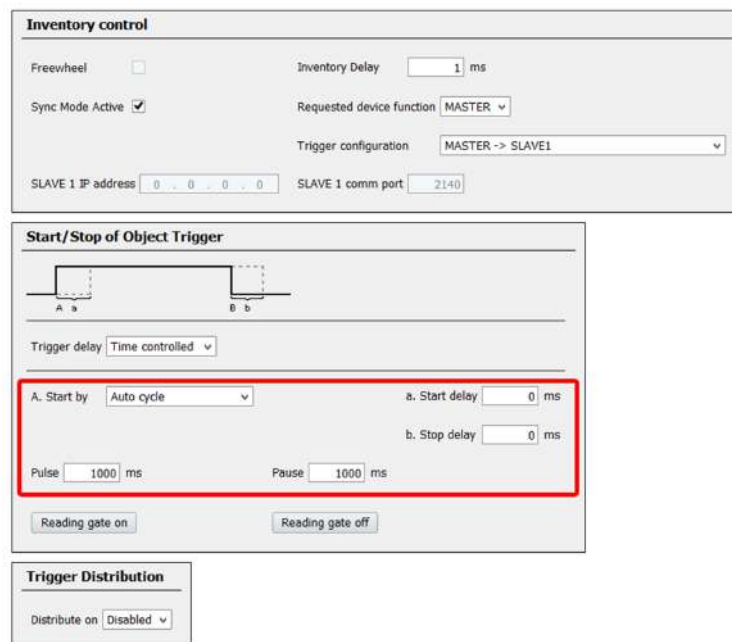


Figura 4.5: SOPAS - Temporização das aquisições

4.1.5 Formatação dos Dados de Saída

Os leitores, depois de lerem as etiquetas, informam que só podem ser processadas como saídas, as etiquetas que obedecem a condições específicas (nível de sinal, por exemplo) e formata o que é apresentado como saída. A composição da saída pode ser definida na configuração por uma *string* específica (figura 4.6), com caracteres especiais ou informações da própria *tag* (UUI, memória interna, posições absolutas xyz).

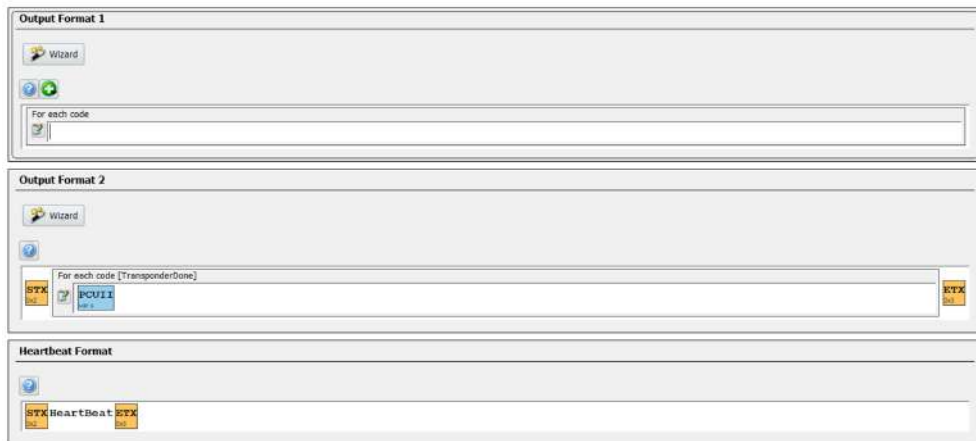


Figura 4.6: SOPAS - formatar output do leitor

De acordo com a figura 4.7 foi necessário criar um protocolo no próprio leitor utilizando UUI para identificar cada *tag* e usá-la como cursor na base de dados, uma vez que a memória interna não tem espaço suficiente para armazenar todo o volume de dados. As figuras 4.7 e 4.8 mostram como adicionar a *string* de saída ao UUI e como configurá-la de modo a que os dados sejam apresentados em ASCII, caso contrário os dados seriam enviados em *hex*, o que exigiria um processamento para obter o nome da etiqueta e permitir visualizar o resto do sistema.

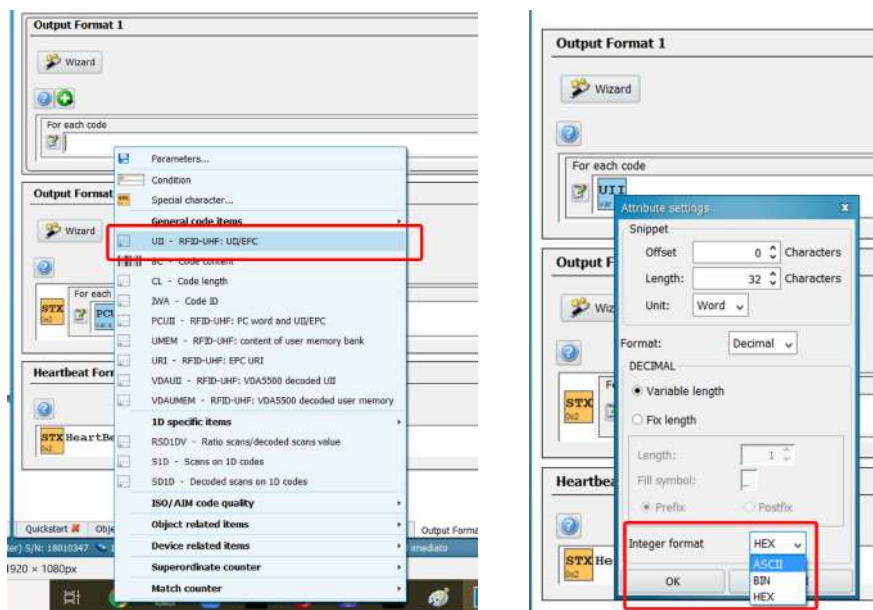


Figura 4.7: SOPAS - Selecionar Formatar Figura 4.8: SOPAS - Formatar UUI baseado na tabela ASCII

4.1.6 Configurações de Conexão - Ethernet

A ligação aos leitores é realizada por *ethernet* e isto implica que o *hardware* tenha endereços de acordo com o protocolo *TCP/IP*, que podem ser alterados (figura 4.9). Também é possível alterar o modo de funcionamento na rede, seja servidor ou cliente, definindo assim se o próprio leitor inicia ligações a um servidor, como cliente, ou se espera que os outros dispositivos se conectem a ele, como servidor.

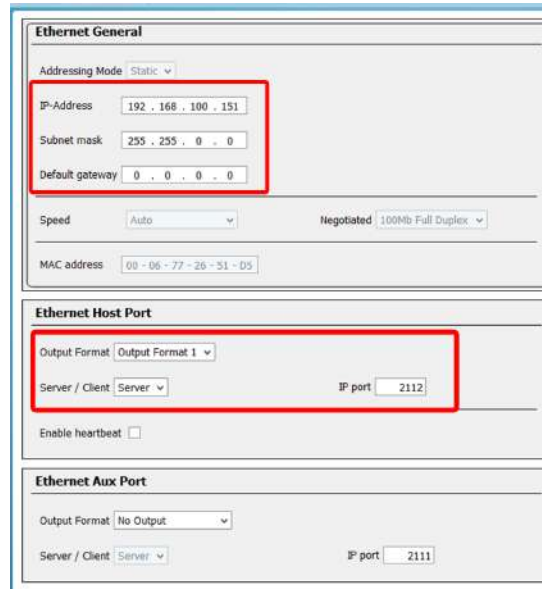


Figura 4.9: SOPAS - Endereço TCP/IP e modo de trabalho

4.1.7 Gravação de Novos Parâmetros

Para todas as alterações acima, deve registar-se com o leitor para as fazer. Os novos dados são enviados para a fila de escrita clicando com o lado esquerdo do rato na propriedade alterada (figura 4.10).

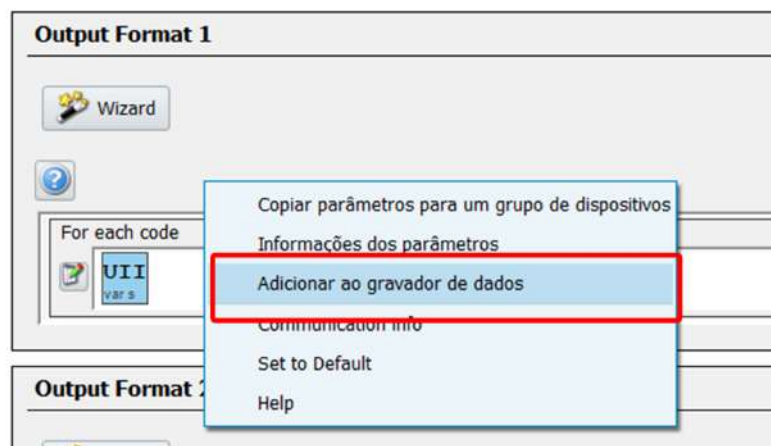


Figura 4.10: SOPAS - Adicionar a fila de gravação novos parâmetros

Nas figuras 4.11 e 4.12 procura detalhar-se que os parâmetros podem ser escritos de duas formas: temporários ou permanentes. Os primeiros parâmetros são enviados e podem ser utilizados até que o leitor seja desligado e quando religado os parâmetros anteriores são

carregados. O segundo salva permanentemente todos os parâmetros mantendo-os carregados, mesmo que o leitor seja desligado mais tarde.

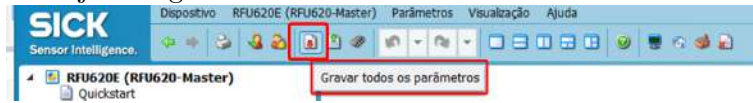


Figura 4.11: SOPAS - gravar os novos parâmetros



Figura 4.12: SOPAS - Gravar os novos parâmetros de forma permanente

4.2 Autômato Programável (PLC) e Painel de Operação (HMI)

No trabalho implementado utilizou-se o PLC Siemens S7-1200 (CPU 1214AC/DC/Relay) [113], versão integrada que apresenta 14 entradas digitais, 10 saídas digitais (por relé), 2 entradas analógicas de 0-10 volts com 10 bits de resolução e comunicação *profinet*. Para visualização dos processos foi utilizado o painel Siemens KTP600 (6AV6 647-0AD11-3AX0) [114], com *display* de 6", TFT de 256 cores, *touchscreen* e comunicação *profinet*.

Finalmente utilizou-se um sistema de supervisão SCADA-*Supervisory Control and Data Acquisition* da *Siemens* [115] com o SIMATIC PC STATION - WINCC RT ADVANCED. Todos os dispositivos citados podem ser configurados e programados pelo *software Siemens TIA PORTAL V15* [116].

4.2.1 Estrutura da Rede Industrial

Como pode ser visto na figura 4.13 a estrutura da rede industrial é basicamente composta por conexões em *profinet (TCP/IP)* definidas em *LAN* pelos endereços: PLC - 192.168.100.110, HMI - 192.168.100.2 e SCADA - 192.168.100.100

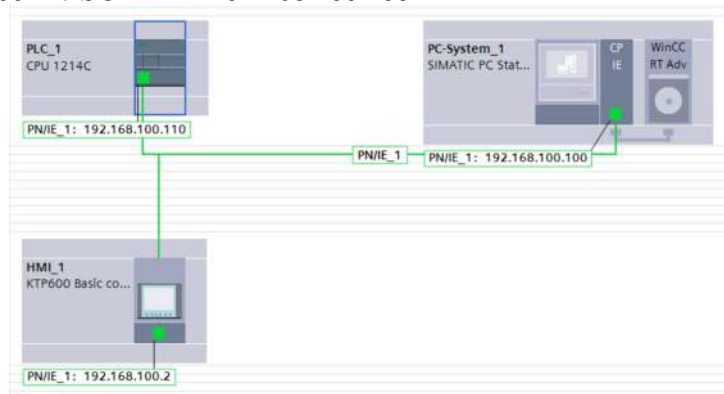


Figura 4.13: TIA portal - Rede Industrial

4.2.2 Programação desenvolvida - S7 1200

O fluxo de trabalho do PLC S7-1200 (Figura 4.14) é realizado iniciando as conexões com o sensor RFID, com o HMI e com o SCADA. Após obter todas as conexões, o PLC fica à espera de receber novos dados do sensor e quando isto acontece faz o processamento, executando as funções: *Function Block 2* e *Function Block 3* (ver fig. 4.14). Uma vez os dados processados, são enviados para visualização do utilizador, no HMI e no SCADA. Por fim, verifica-se ainda se existe a conexão com o sensor RFID e, caso exista, volta a esperar receber novos dados para fazer processamento, até que a conexão com o sensor seja interrompida, por quedas de ligação ou mesmo pelo utilizador e, deste modo, não havendo mais conexão, é terminado o programa.

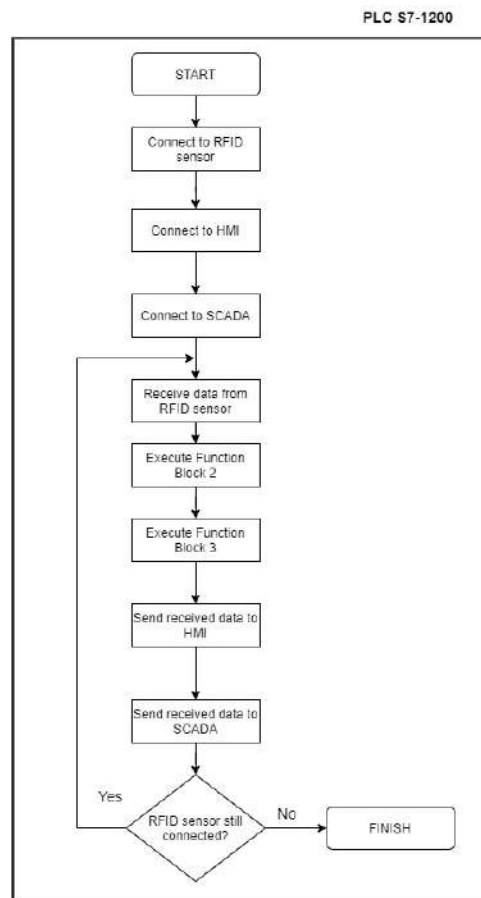


Figura 4.14: Fluxograma de trabalho do S7-1200

O PLC S7-1200 realiza a conexão direta, via *ethernet*, com o sensor RFID, recebendo e processando os dados para que sejam enviados para a Base de Dados *online*. Para executar a comunicação *ethernet* do PLC, o TIA PORTAL [116] disponibiliza a função “TRCV_C”. Na figura 4.15, é exibida a caixa de configuração, onde é especificado o tipo de conexão (*TCP* ou *ISO-on-TCP*), o identificador da comunicação (ID), qual dispositivo a iniciar a conexão (ex: se é o PLC a iniciar a conexão com o outro dispositivo ou se o PLC espera ser conectado). Informações adicionais encontram-se em anexo.

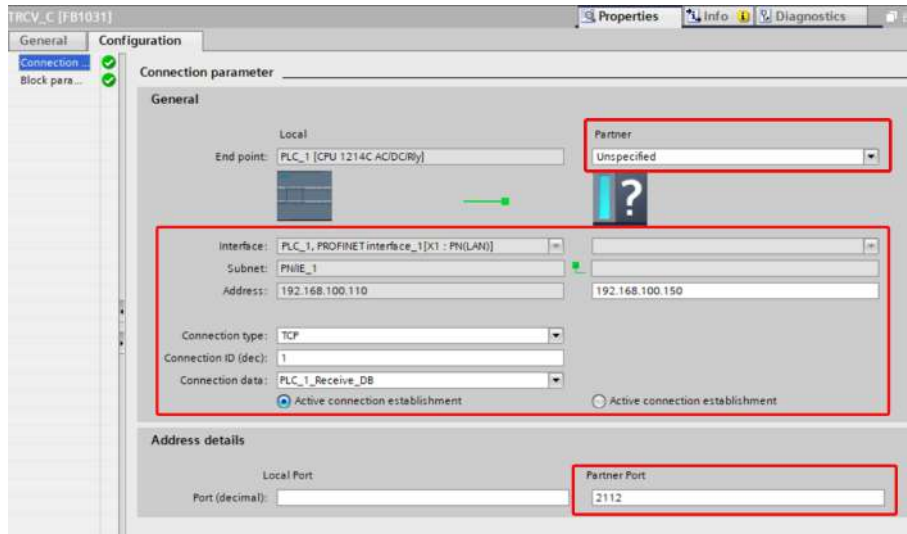


Figura 4.15: TIA PORTAL - TRCV_C

O fluxograma da figura 4.16, é responsável por processar os dados recebidos e retirar os caracteres do protocolo. Inicialmente, os dados recebidos que estão em *array de char* são convertidos em string pela função *Chars to Strg*. Em seguida, utilizando a função *FIND* procura-se a posição de cada caracter do protocolo. Por fim, com a condição IF verifica-se a diferença entre as posições, para garantir que existam dados dentro do protocolo e por fim, com a função *MID* extraem-se os dados, excluindo o protocolo. A figura 4.17 é a função “BLOCK 2” em linguagem SCL (*Structured Control Language*).

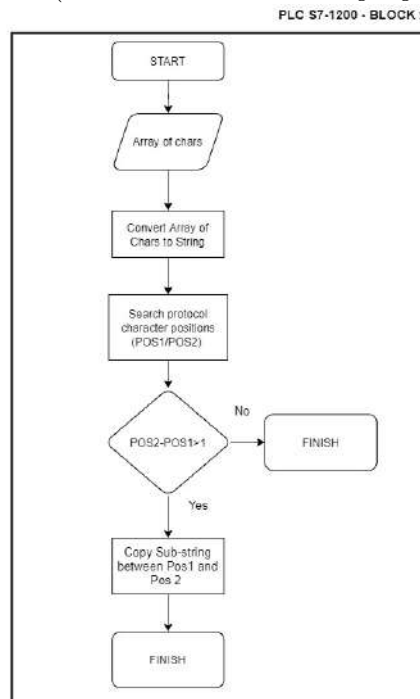


Figura 4.16: Fluxograma do Block 2

```

0001 Chars_TO_Strg(Chars:=#Dados_array,
0002                pChars:=0,
0003                Cnt:=0,
0004                Strg=>#string_temp1);
0005
0006 #pos1 := FIND(IN1 := #string_temp1, IN2 := '*');
0007 #pos2 := FIND(IN1 := #string_temp1, IN2 := '#');
0008
0009 IF((#pos2-#pos1)>1) THEN
0010     #Dados_string := MID(IN := #string_temp1,
0011                          L := (#pos2 - #pos1)-1,
0012                          P := #pos1 + 1);
0013 END_IF;
0014

```

Figura 4.17: TIA PORTAL - BLOCK 2

O fluxograma da figura 4.18, é apenas uma verificação se a string recebida é diferente da anterior. Caso a condição seja Verdadeira, faz-se uma copia da atual e envia-se um bit como verdadeiro. Caso a condição seja Falsa, é assinalado o bit como falso. Na figura 4.19 ilustra-se o código, em linguagem SCL, da função “BLOCK 3”.

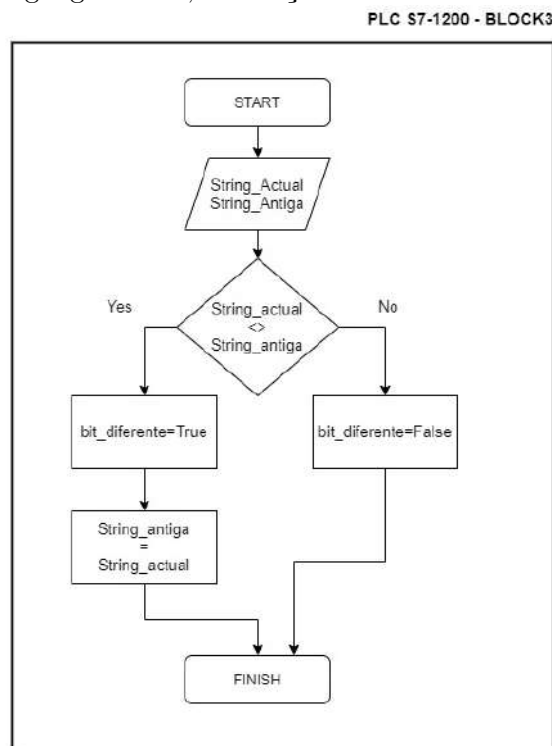


Figura 4.18: Fluxograma do Block 3

```

0001
0002 IF #String_atual <> #String_antiga THEN
0003     #bit_diferente:=TRUE;
0004     #String_antiga := LEFT(IN := #String_atual, L := LEN(#String_atual));
0005 ELSE
0006     #bit_diferente := FALSE;
0007 END_IF ;
0008

```

Figura 4.19: TIA PORTAL - BLOCK 3

A figura 4.20 exhibe o código, escrito em linguagem ladder, encarregado de chamar as funções de comunicação (TRCV_C), Block 2 e Block 3.

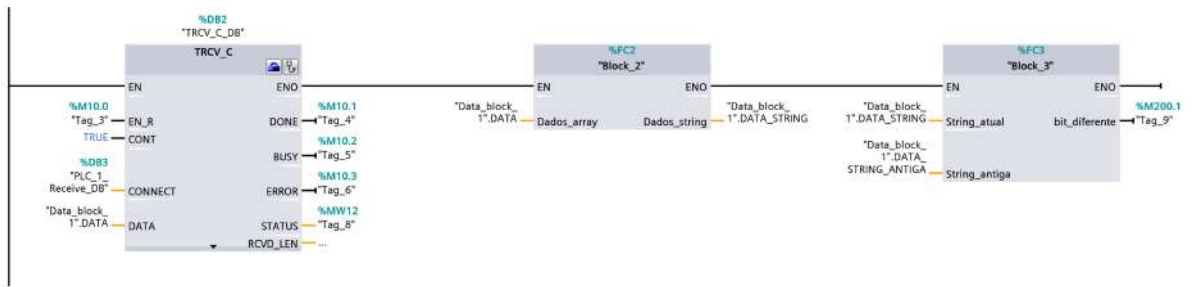


Figura 4.20: TIA PORTAL - BLOCK 1

4.2.3 Programação do Painel HMI

O Painel HMI é composto basicamente por botões e campo de entrada/saída, como mostrado na figura 4.21. Os itens 1 e 2 são os botões para habilitar/desabilitar a conexão e o item 3 é o campo de entrada/saída que é utilizado para visualização do processo.

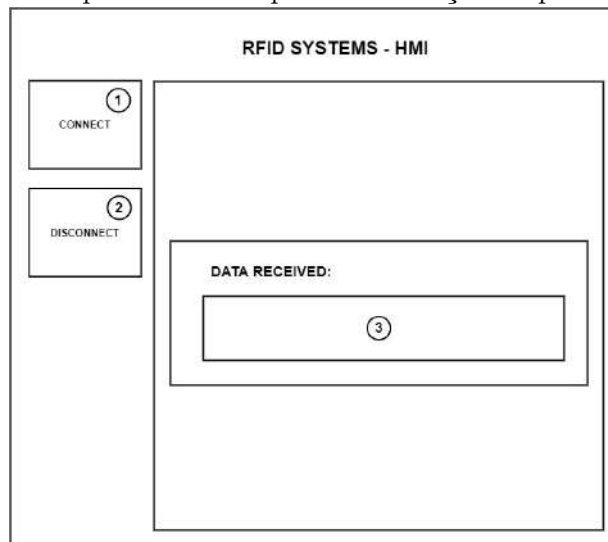


Figura 4.21: TIA PORTAL - Tela da HMI e seus itens

O controle efetuado pelos botões é atribuído quando se altera as propriedades e se cria o evento de *set bit* e *reset bit*. No caso do botão para habilitar a conexão, atribui-se o valor da *tag* associada à função como verdadeiro, e na desabilitação da conexão o botão anula (reset) o valor da *tag*, atribuindo assim o valor falso. Na figura 4.22 expõem-se as propriedades para configuração do botão de conexão, atribuindo a função “*SetBit*” da *tag tag_3* ao evento de *click*.

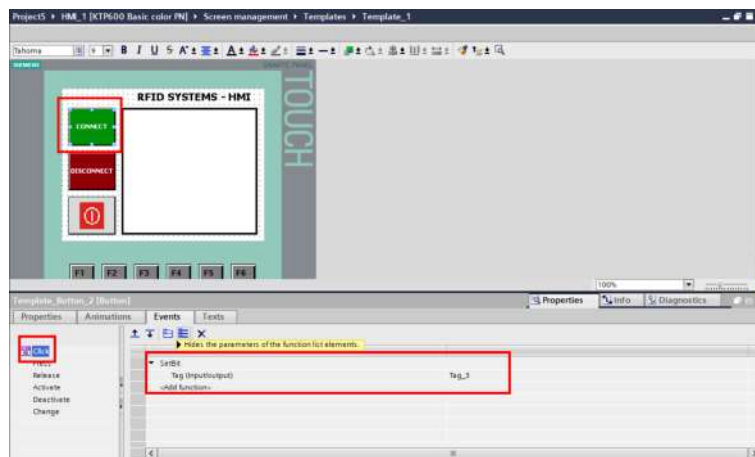


Figura 4.22: TIA PORTAL - Função *SetBit* ao evento click

A configuração de campos de entrada/saída é semelhante à atribuição de eventos a botões porém, torna-se mais simples pois nas próprias propriedades gerais dos campos existe uma possibilidade para atribuir a *tag* que vai ser monitorizada. Destaca-se na figura 4.23 a atribuição da *tag* “Data_block_1_DATA_STRING” ao objeto “IO_field.1” no formato de string.

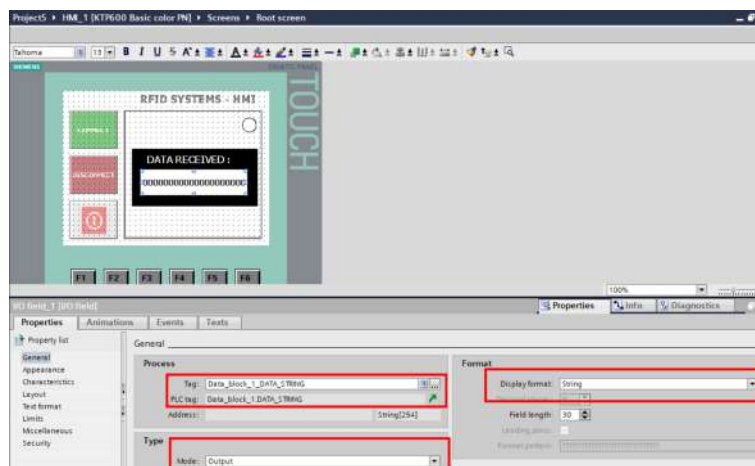


Figura 4.23: TIA PORTAL - Atribuição da *tag* ao campo de entrada/saída

4.3 Sistema de Supervisão e Aquisição de Dados - SCADA

4.3.1 Configuração PG/PC

A configuração necessária para aceder ao SCADA-PC, tanto para programá-lo como para lhe aceder como aplicação SCADA, é definida pelo PG/PC *software* instalado automaticamente no TIA PORTAL V15. Para aceder, deve abrir-se o painel de controlo do *Windows* e pesquisar por PG/PC (figura 4.24). Em seguida, abre-se-nos uma janela com a lista de dispositivos disponíveis para seleccionar como “ponto de acesso”.

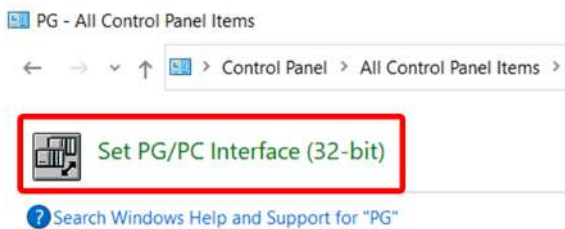


Figura 4.24: SCADA - Painel de Controlro

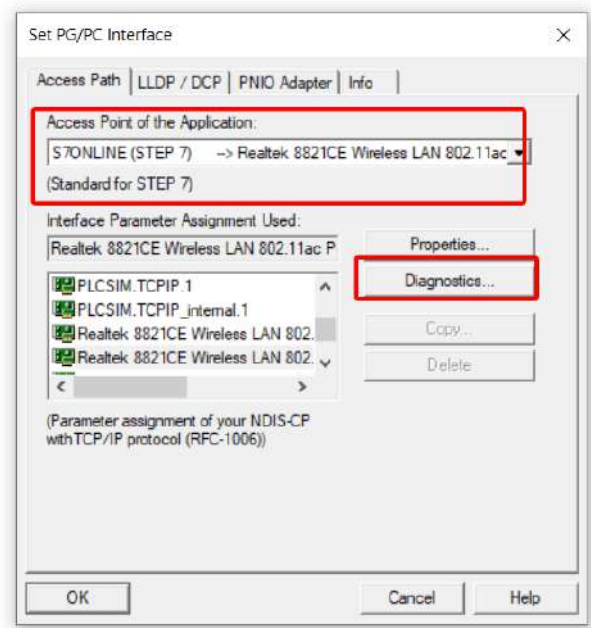


Figura 4.25: SCADA - Interface PG/PC

Para verificar a conexão é aconselhável realizar o diagnóstico do “ponto de acesso”, com o dispositivo selecionado. Carrega-se em Diagnostics (figura 4.25) e depois em Test (figura 4.26). No TIA PORTAL verificam-se os IPs (do PLC e do PC-SCADA) e por fim, verifica-se que a conexão está com o ponto de acesso correto (figura 4.27).

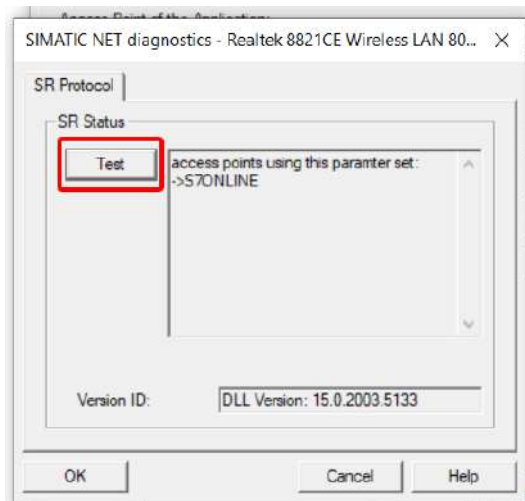


Figura 4.26: SCADA - Diagnóstico PG/PC

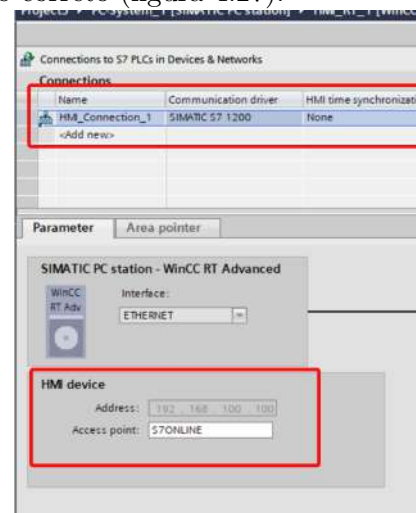


Figura 4.27: SCADA - Conexões SCADA

4.3.2 Configuração de SCRIPTS/TAGS/BOTÕES

O sistema SCADA permite executar diversas funções como carregar em botões, animar algo para que seja perceptível visualmente, salvar dados em Excel e até executar códigos em linguagem C e VBscript. No caso do trabalho implementado foram desenvolvidos 2 códigos em VBscript com as tarefas de inserir registos provenientes dos sensores RFID e inseri-los como novos objetos na Base de Dados *SQL SERVER* (figura 4.28) [117]. A chamada desses códigos é feita quando as tags *Data_block_1.DATA_STRING* e *Data_block_1.DATA_INSERT_STRING*

mudam seu valor atual, acionando assim o evento "value change"(figura 4.29).

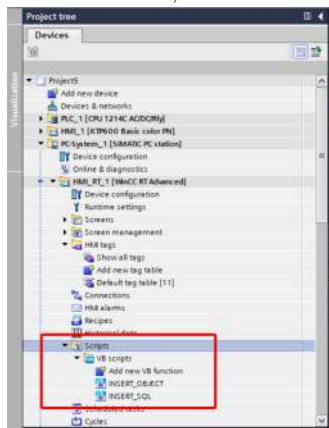


Figura 4.28: SCADA - Lista de *scripts*

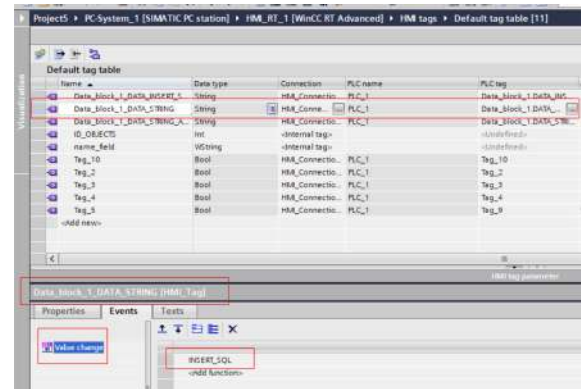


Figura 4.29: SCADA - Evento *value change*

A configuração de botões e campos de texto são semelhantes à chamada de *scripts*, porém a associação de *tags* e os seus eventos são realizados directamente no ecrã desenvolvido para o SCADA e não na tabela de *tags*. De acordo com a figura 4.30, ao associar um evento a um botão é basicamente seleccioná-lo, verificando qual o evento que melhor se adequa e qual a função que deve ser executada, sendo um *script*, ou *set/reset* de um *bit* de dados, associados à *tag*.

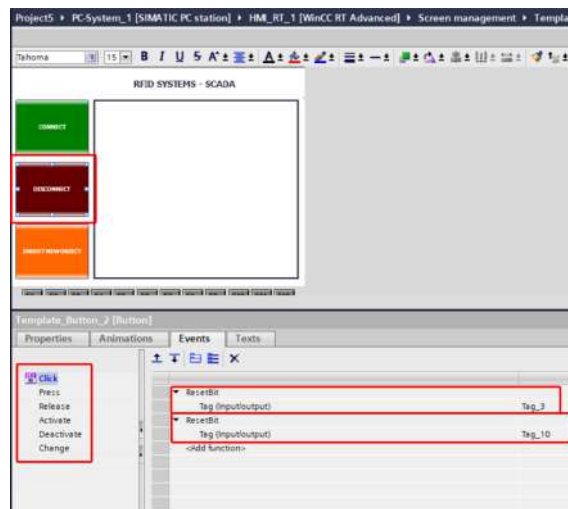


Figura 4.30: SCADA - Aba eventos dos botões

Para o processo de selecção de *tags*, na aba das propriedades gerais, há um campo para seleccionar a *tag*. Dependendo do modo de operação do campo (saída, entrada ou entrada/saída), poderemos apenas visualizar os seus dados (*Output field*), ou escrever novos dados (*Input field*). Para a parametrização de campos de texto (figura 4.31) é um processo bastante mais simples, apenas seleccionando esse campo no ecrã desenvolvido e escrevendo o texto que se pretende.

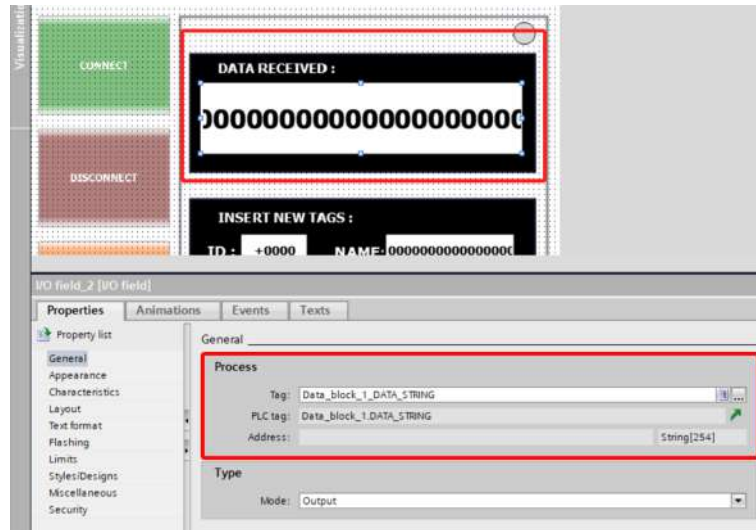


Figura 4.31: SCADA - Selecionar *tag* para *inputs/outputs*

O SCADA desenvolvido no presente trabalho pode ser ilustrado na figura 4.32, cuja sua principal estrutura apresenta botões, campos de entrada e execução de *scripts* em *Visual Basic Script (VBscript)*. Estes campos são identificados na figura, com a numeração de 1 a 6, de modo a serem, seguidamente, sumariamente caracterizados.

O item 1 é o botão responsável por dar início à conexão, atribuindo o valor lógico à *tag* de conexão de verdadeiro. O item 2 é o botão responsável por fechar a conexão, atribuindo o valor lógico à *tag* de conexão de falso. O item 3 é o botão cuja função é registar novos objetos, logo a conexão iniciada por ele, apenas regista objetos novos. O item 4 é um campo de entrada/saída de dados, o qual é apenas utilizado para visualização de varrimentos dos sensores RFID e de cada vez que há um novo objeto, ativa-se o evento da *tag* “*value change*” e executa-se o *script*, em VBscript, para envio dos dados para a Base de Dados. Os itens 5 e 6 são os campos de registo de novos objetos no sistema, sendo o item 5 o código de registo e item 6 o *UII* do objeto registado.

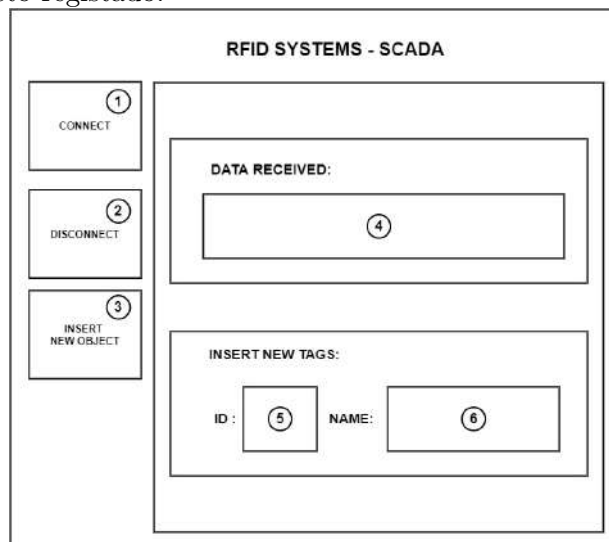


Figura 4.32: Ecran principal do SCADA

4.4 Interfaces Gráficas - PC e DATATRANSFER

As interfaces gráficas foram desenvolvidas em IDE Rad Studio 10.4 [118] e em linguagem Delphi [119]. Na figura 4.33, é mostrado o fluxograma de trabalho da aplicação PC. Uma vez iniciada a aplicação, procura-se estabelecer-se a conexão com o sensor RFID e com a Base de Dados online. Após a realização desse procedimento, a aplicação aguarda a recepção de dados provenientes do sensor. Ao recebê-los, extrai-se o identificador único universal das *tags* (UUI - *Universal Unique Identifier*) e realiza-se a concatenação do UUI com os dados de Data-Hora-Área, e envia-se para a Base de Dados online a *string* “DATA - HORA - UUI - AREA”. Por fim, o programa verifica se o sensor RFID e a Base de Dados online ainda estão conectados e, em caso afirmativo, o fluxo de trabalho retorna à fase inicial de recepção de dados do sensor RFID. Caso a conexão não esteja mais estabelecida, o programa fica à espera que o utilizador realize a operação de *reset*.

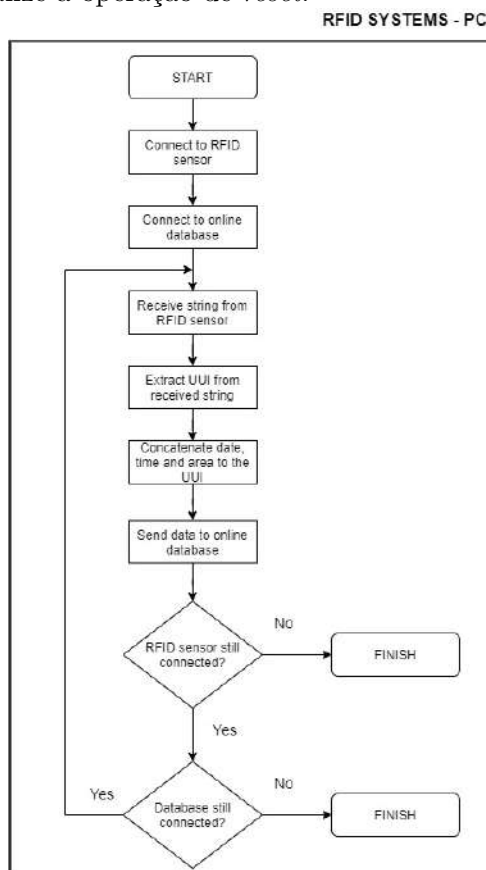


Figura 4.33: Fluxograma da Aplicação PC

Nas figuras 4.34 e 4.35 são representados os ecrãs de conexão com o sensor RFID e a Base de Dados online, respetivamente. A conexão com o sensor é criada a partir do IP/Porta/Área. Por princípio, a conexão *TCP/IP*, exige apenas IP/Porta, no entanto, para que o programa identifique em que área o sensor está em funcionamento será necessário proceder-se à sua inserção previamente. Isso garante a escalabilidade da aplicação desenvolvida, uma vez que permite, ler diferentes sensores RFID localizados nas diversas fases da cadeia de valor do produto rastreado (neste estudo, blocos de mármore).

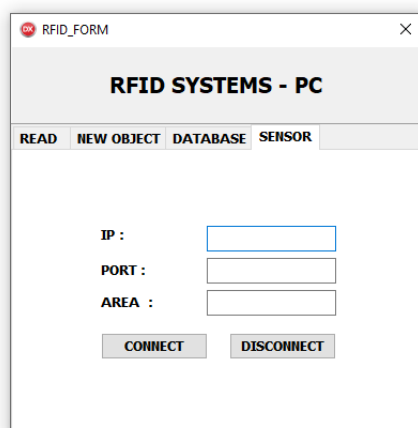


Figura 4.34: PC local – Conexão do sensor RFID

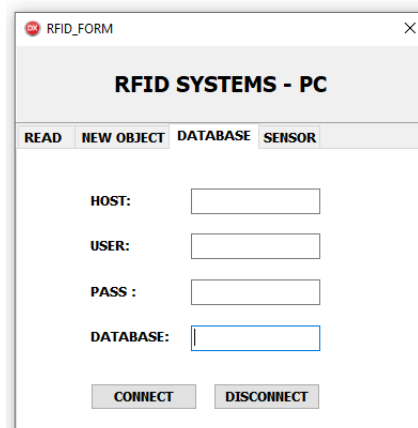


Figura 4.35: PC local - Conexão com a Base de Dados *online*

Nas figuras 4.36 e 4.37 apresentam-se os ecrãs: Terminal de receção de dados; Registo de novos objetos. O ecrã do terminal destina-se apenas à visualização/*debug* dos dados recebidos pelos sensores RFID. O ecrã de Registo de novos objetos recebe como entrada o ID do objeto (número de identificação) e o nome do objeto (UUI). Também é possível a visualização dos objetos registados na Base de Dados *online*.

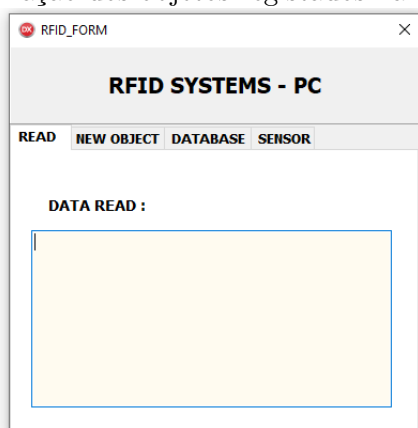


Figura 4.36: Terminal de entradas recebidas do sensor RFID

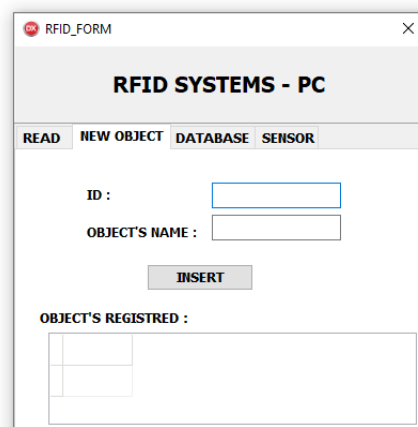


Figura 4.37: PC local - Registo de novos objetos

Na figura 4.38, é apresentado o fluxograma do programa *DATATRANSFER*, esta aplicação tem a função de transferir dados entre duas Bases de Dados, dado não ser possível a transferência directa, em função das Bases de Dados utilizadas neste trabalho: *MySQL* [120] e *SQL SERVER* [117]. O fluxo começa inicializando as conexões com as Bases de Dados, *MySQL* e *SQL SERVER*, após estabelecida a conexão, a aplicação lê a última linha da tabela de entradas do *SQL SERVER* e do *MYSQL* e verifica se são iguais. Caso haja a igualdade nos dados de ambas as Bases de Dados, o programa volta a ler a última linha das tabelas até que os dados sejam diferentes. No caso dos dados serem diferentes, nas duas linhas lidas das tabelas, o programa insere os dados do *SQL SERVER* no *MYSQL*. No final do programa são verificadas as conexões com as Bases de Dados. Se as conexões se mantiverem *online*, o programa volta a receber e verificar os dados das duas Bases. Se a conexão deixar de existir, o programa espera que o utilizador execute as operações de *reset*.

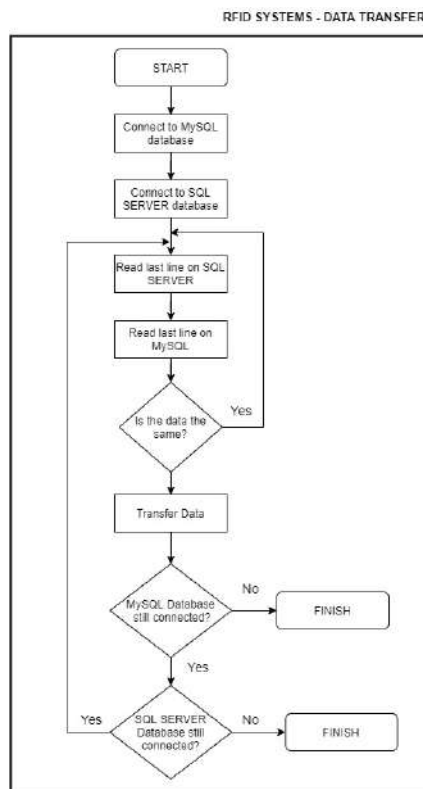


Figura 4.38: Fluxograma da Aplicação *DATATRANSFER*

A figura 4.39 corresponde ao ecrã inicial do programa *DATATRANSFER*, no qual é possível visualizar os dados recebidos pelas Bases de Dados e onde se iniciam as verificações entre as duas Bases. A existência da informação “área” permite a escalabilidade da aplicação desenvolvida, uma vez que permite, ler diferentes sensores RFID localizados nas diversas fases da cadeia de valor do produto rastreado (conforme fora atrás já referido).

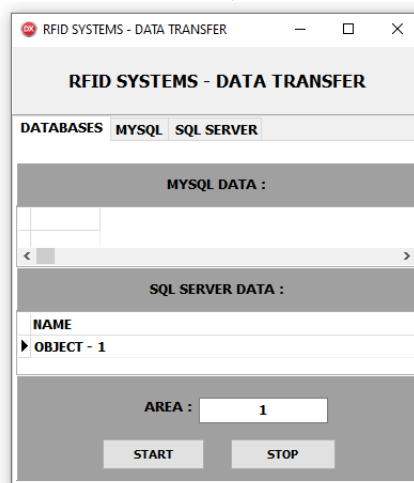


Figura 4.39: Ecrã Inicial da aplicação *DATATRANSFER*

Finalmente as figuras 4.40 e 4.41, ilustram as abas de conexão com as bases de dados e apresentam os dados de conexão, nomeadamente: *host* (endereço da base), nome do utilizador registado na Base, senha e nome da Base de Dados que está a ser acedida.

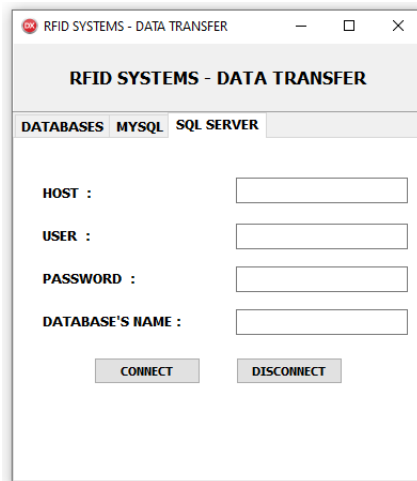


Figura 4.40: Ecrã de conexão da Base de Dados - *MySQL*

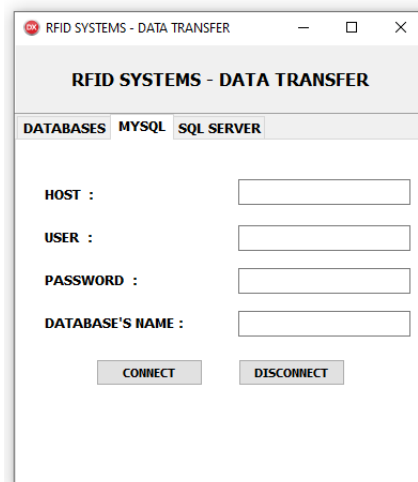


Figura 4.41: Ecrã de conexão da Base de Dados - *SQL SERVER*

4.5 Plataforma Web

A plataforma *online* é constituída pelas secções: *scans* dos sensores *RFID*; objetos; áreas; pesquisa de objetos e arquivos de objetos. Esta Plataforma *Web* foi desenvolvida com recurso às linguagens *HTML5* [121], *PHP* [122] e *CSS* [123], com o recurso do framework *Bootstrap* [124].

A secção relativa aos *scans* dos sensores *RFID* tem a finalidade de visualização de quais as *tags* que foram lidas pelos sensores e disponibilizam a seguinte informação: HORA - DATA - OBJECTO - ÁREA (ver figura 4.42).

TABLE FROM AREA 100-SYSTEM				TABLE FROM AREA 100-SYSTEMS			
TIME	DATE	OBJECT	AREA	TIME	DATE	OBJECT	AREA
18:40:59	2020-11-26	OBJECT-10000	1	18:41:13	2020-11-26	OBJECT-10000	2

Figura 4.42: PLATAFORMA WEB – Visualização de scans dos sensores RFID

A secção objetos (figura 4.43) permite a visualização dos objetos registados na Base de Dados e acrescentar ou suprimir objetos.

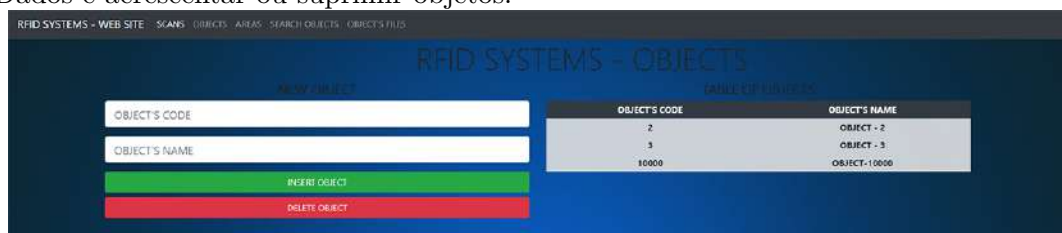


Figura 4.43: Plataforma WEB – Visualização de Objectos

A secção Áreas (figura 4.44) permite a visualização das áreas registadas na Base de Dados e acrescentar ou suprimir áreas.



Figura 4.44: Plataforma WEB – Visualização de Áreas

Na secção Arquivos dos Objetos (figura 4.45) é permitida a visualização dos arquivos anexados a cada Objeto, na Base de Dados, e e acrescentar ou suprimir arquivos.

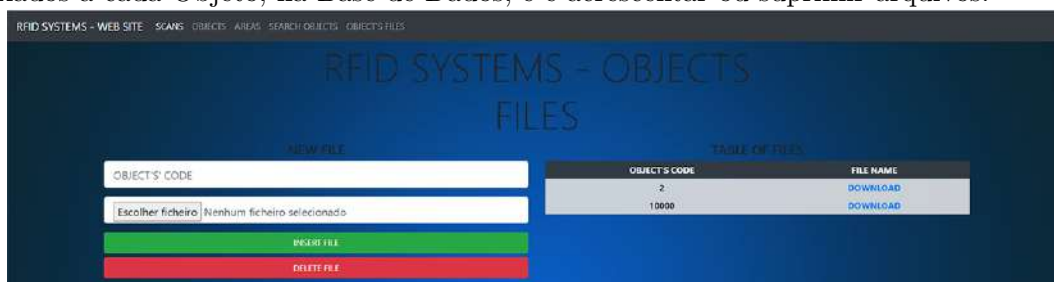


Figura 4.45: PLATAFORMA WEB - EXIBIR/APAGAR NOVOS ARQUIVOS

Na secção Pesquisa de Objetos (figuras 4.46 e 4.47) é permitida a realização de busca por Objetos registados e dos seus dados anexos. Por exemplo, um Objeto que tenha passado por diversas áreas, terá vários arquivos anexados, e como resultado da busca é exibido todo o seu histórico de passagens pelo sistema, com a possibilidade de *download* dos seus arquivos.

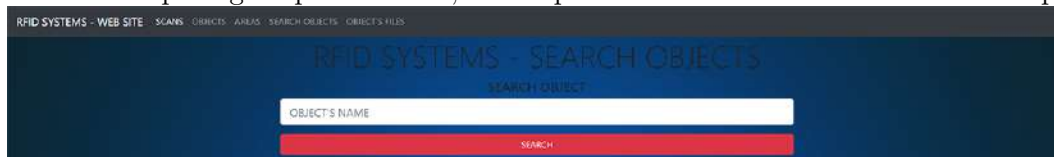


Figura 4.46: PLATAFORMA WEB - BUSCA DE OBJECTOS



Figura 4.47: PLATAFORMA WEB - RESULTADO DA BUSCA

4.6 MYSQL e SQL SERVER

A linguagem utilizada para gerir a base de dados é a *SQL (Structured Query Language)* [125] amplamente aplicada em bases de dados relacionais, nas tarefas de criação da base e suas tabelas, inserção de dados e consulta. O trabalho realizado implementou duas bases de dados: uma em *MySQL* [120] e outra em *SQL SERVER* [117]; que, apesar de não se comunicarem diretamente, ambas utilizam a linguagem *SQL*, embora cada uma com suas especificações. Abaixo segue uma lista dos comandos mais utilizados.

- **SELECT:** Consultar dados de uma tabela .

- **DELETE:** Apagar dados de uma tabela .
- **INSERT INTO:** Inserir dados em numa tabela .
- **CREATE DATABASE/TABLE:** Criar a base de dados / Criar tabela numa base de dados.
- **DROP DATABASE/TABLE:** Apagar base de dados/ Apagar tabela numa base de dados

4.6.1 *MYSQL* - Estrutura dos dados

Na criação de tabelas deve verificar-se que tipo de dados estão a ser inseridos para que possam ser armazenados da melhor forma possível, evitando assim perdas de precisão ou alocação de espaços desnecessários na base. A base de dados em *MySQL* está *online* e deste modo pode ser administrada diretamente por ferramentas disponíveis pela empresa de hospedagem ou programas gerenciadores de bases de dados. No presente trabalho utilizou-se o *software open-source HeidiSQL* [126] para realizar operações de criação de tabelas, inserção de dados e consulta de tabelas.

4.6.2 *SQL SERVER* - Configuração do ODBC

Para que a base de dados em *SQL SERVER* seja acessível para os programas do *Windows*, deve primeiramente configurar-se o *ODBC* (*Open Database Connection*), abrindo-se o painel de controlo do *Windows* e pesquisando por *ODBC* (figura 4.48), em seguida seleciona-se a janela conforme o sistema dados utilizados (32-bits ou 64 bits). Com a janela do *ODBC* aberta (figura 4.49) é possível ver a lista de bases de dados visíveis pelo sistema e a opção para adicionar novas bases de dados à lista.

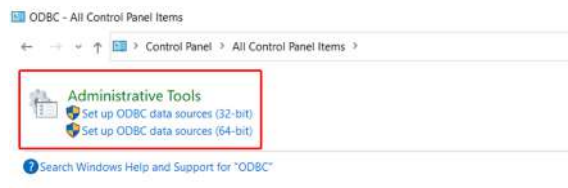


Figura 4.48: *SQL SERVER* - Painel de Controlo *ODBC*

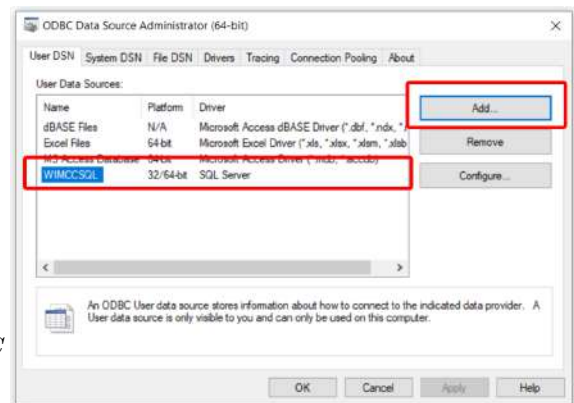


Figura 4.49: *SQL SERVER* - Lista de Bases *ODBC*

Nas figuras 4.50 e 4.51 mostram-se, respectivamente, a lista de bases de dados com suporte *ODBC* e os dados do novo acesso.

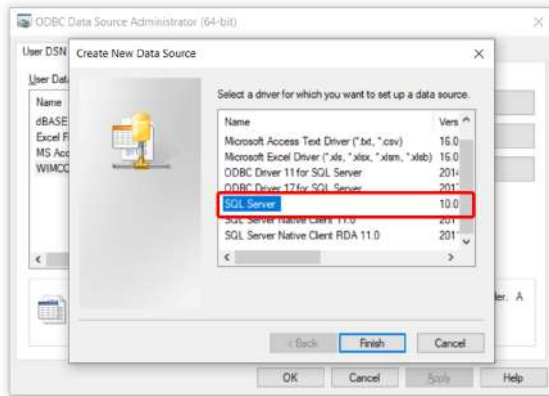


Figura 4.50: *SQL SERVER* - Bases de dados com suporte *ODBC*

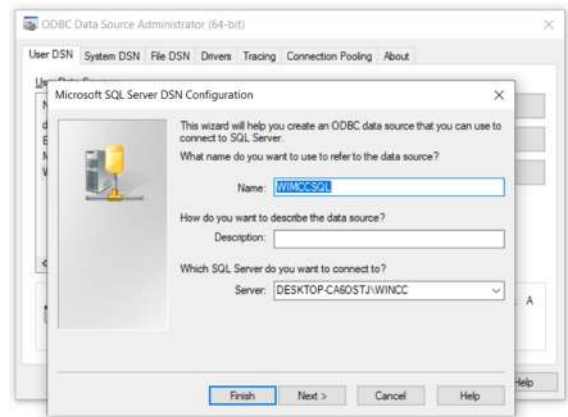


Figura 4.51: *SQL Server* -Dados do novo acesso

Na figura 4.52 é verificado se o acesso é feito através de um utilizador do *Windows* ou de um utilizador criado dentro da base de dados. Na figura 4.53 exemplifica-se a confirmação de qual base *SQL SERVER* é seleccionada para se ter acesso. No presente trabalho foi implementada a "WINCC.TEST".

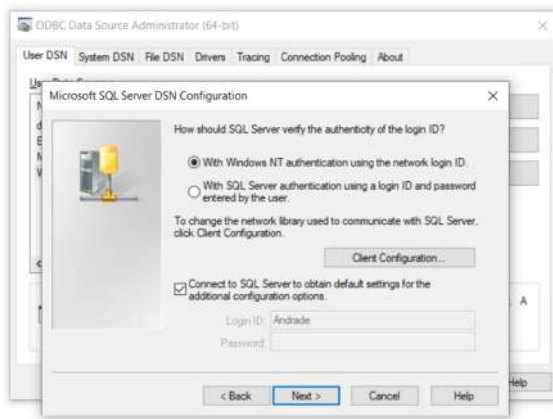


Figura 4.52: *SQL SERVER* - Autenticação do acesso à base

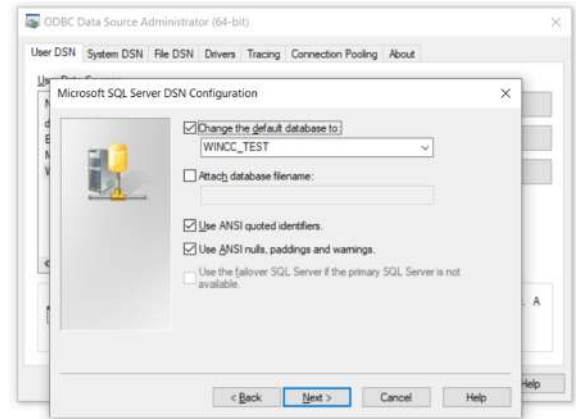


Figura 4.53: *SQL SERVER* - Escolha da base de dados

Para finalização do processo de criação do acesso *ODBC* para a base de dados em *SQL SERVER* são exibidos os dados de configuração (figura 4.54) assim como a opção para testar o acesso à Base de dados (figura 4.55).

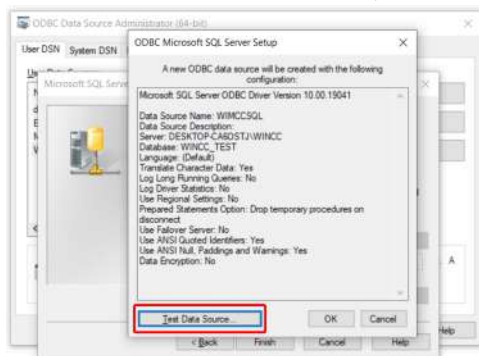


Figura 4.54: *SQL SERVER* - Dados do acesso ao *ODBC*

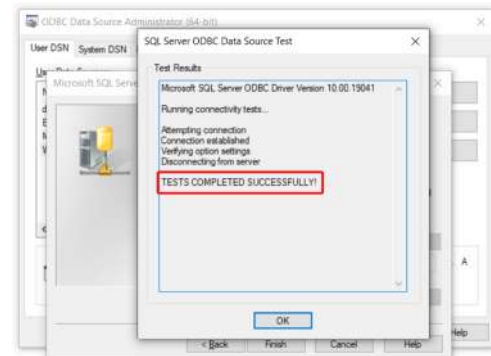


Figura 4.55: *SQL SERVER* - Status do acesso ao *ODBC*

5 Conclusões e Sugestões para Trabalhos Futuros

Nesta dissertação apresentou-se uma estratégia de implementação que é vista como um ativo valioso para a indústria do mármore, pois demonstra através de um protótipo de prova de conceito, como a integração de sistemas RFID e plataformas web são atores fundamentais para apoiar a conectividade entre os parceiros de negócio da cadeia produtiva. A utilização de sistemas RFID juntamente com as plataformas Web, desenvolvidas, trazem um novo contexto de digitalização onde todos os parceiros (clientes finais, engenheiros, arquitetos) poderão escolher produtos de mármore adequados, de acordo com suas superfícies visuais, ou propriedades elétricas, antes mesmo do produto final ser cortado do bloco de pedra.

O trabalho realizado nesta dissertação foi objecto de uma publicação em revista internacional [25] e atingiu os seus objetivos principais ao aproximar a indústria do mármore do novo paradigma da indústria 4.0, permitindo rastreabilidade automática dos produtos finais, facilitando os processos de armazenamento, o controlo de stocks/ vendas e integrando, numa plataforma digital, todos os participantes da cadeia produtiva da indústria do mármore.

Para possibilitar esta implementação, foram utilizadas as linguagens de programação Delphi [119], VBscript [127], SCL [128], SQL [125], PHP [122], HTML [121] e BootStrap [124]. A escolha da linguagem Delphi [119] deve-se fundamentalmente à sua fácil implementação de GUIs (Graphical User Interface), conexões TCP/IP e comunicações com base de dados, o que tornou a sua utilização uma parte importante e significativa no projeto. A utilização do software TIA PORTAL V15 [116] no desenvolvimento do protótipo industrial, possibilitou maior integração, tendo em vista a configuração e a programação das entidades: PLC S7-1200 [113], HMI [115] e sistema SCADA [114], na mesma plataforma. A implementação das bases de dados com a linguagem SQL foi encarada como eficiente, devido à sua rapidez e fiabilidade no processamento de grandes quantidades de dados.

É importante salientar que as aplicações desenvolvidas tiveram em conta dotar o sistema de flexibilidade relativamente à sua potencial escalabilidade futura. É possível a selecção da aplicação RFID-PLC (quando se trabalha em contexto fabril, com PLCs) ou a selecção da aplicação RFID-PC (quando se trabalha em contexto de escritório ou armazém, com PCs). Outra possibilidade é coexistirem sistemas mistos, onde no mesmo local se poderão utilizar diversas tecnologias com a interacção das aplicações RFID-PC e RFID-PLC, simultaneamente, com a selecção da mesma identificação de área.

Por fim, a forma de digitalizar novas informações científicas no domínio da indústria de extração de mármore, onde a identificação RFID é o fator chave para reunir os testes de laboratório científico com os bancos de dados online necessários para permitir a otimização dos processos de corte e transformação.

Para trabalhos futuros, prevê-se a concepção de um sistema de apoio à decisão, totalmente automatizado, para o corte da pedra. O sistema terá a capacidade de sugerir planos de corte com base em critérios para otimizar material, energia e outros custos de processamento associados aos padrões visíveis na superfície do material, resultantes do corte. Esses padrões serão analisados por inteligência artificial, em bases de dados, com algoritmos inteligentes, a fim de se otimizar todo o potencial económico da indústria da pedra.

Referências

- [1] J. Kagermann, H.; Wahlster, W.; Helbig, “Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0,” tech. rep., Berlin, 2013.
- [2] T. Bauernhansl, M. ten Hompel, and B. Vogel-Heuser, eds., *Industrie 4.0 in Produktion, Automatisierung und Logistik*. Springer Fachmedien Wiesbaden, 2014.
- [3] “Bro-cq - controlo de qualidade de blocos em rochas ornamentais.” <https://www.uevora.pt/investigar/projetos?id=3601>. Accessed: 01-05-2021.
- [4] U. Sendler, ed., *Industrie 4.0*. Springer Berlin Heidelberg, 2013.
- [5] E. Westkämper, D. Spath, C. Constantinescu, and J. Lentjes, eds., *Digitale Produktion*. Springer Berlin Heidelberg, 2013.
- [6] Plattform Industrie 4.0, “Whitepaper FuE-Themen,” no. April, p. 49, 2015.
- [7] M. Hermann, T. Pentek, and B. Otto, “Design principles for industrie 4.0 scenarios,” in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, IEEE, Jan. 2016.
- [8] M. E. P. Hernandez and S. Reiff-Marganiec, “Classifying smart objects using capabilities,” in *2014 International Conference on Smart Computing*, IEEE, Nov. 2014.
- [9] L. Windelband, C. Fenzl, F. Hunecker, and ..., “Zukünftige Qualifikationsanforderungen durch das „Internet der Dinge “in der Logistik,” ... *Dinge in der Logistik ...*, no. January, 2011.
- [10] G. Fortino, A. Rovella, W. Russo, and C. Savaglio, “On the classification of cyberphysical smart objects in the internet of things,” in *Proceedings of the 5th International Workshop on Networks of Cooperating Objects for Smart Cities (UBICITEC 2014) co-located with CPSWeek 2014, Berlin, Germany, Apr 14, 2014* (G. Fortino, S. Karnouskos, and P. J. Marrón, eds.), vol. 1156 of *CEUR Workshop Proceedings*, pp. 86–94, CEUR-WS.org, 2014.
- [11] A. Botthof and M. Bovenschulte, “Die Autonomik als integratives Technologieparadigma,” 2011.
- [12] J. Bildstein, A.; Seidelmann, *Industrie 4.0 in Produktion, Automatisierung und Logistik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2014.
- [13] E. Geisberger and M. Broy, eds., *agendaCPS*. Springer Berlin Heidelberg, 2012.

-
- [14] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*, IEEE, May 2008.
- [15] "VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik, Industrie 4.0 – Begriffe/Terms." <https://www.vdi.de/ueber-uns/presse/publikationen/details/industrie-40-begriffeterms#>. Accessed: 01-05-2021.
- [16] K. Bettenhausen and S. Kowalewski, "Cyber-Physical Systems : Chancen und Nutzen aus Sicht der Automation," *VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik*, no. April, pp. 1 – 12, 2013.
- [17] D. Lucke, D. Gorzig, M. Kacir, J. Volkmann, C. Haist, M. Sachsenmaier, and H. Rentschler, "Strukturstudie "industrie 4.0 fur baden-wuerttemberg"," *Fraunhofer IPA Ministerium fur Finanzen und Wirtshcaft Baden-Wuerttemberg*, pp. 5–55, 2014.
- [18] D. Arnold, *Handbuch Logistik*. Berlin: Springer, 2008.
- [19] O. Herzog, *Intelligente Objekte : technische Gestaltung, wirtschaftliche Verwertung, gesellschaftliche Wirkung*. Berlin: Springer, 2009.
- [20] M. Deindl, *Gestaltung des Einsatzes von intelligenten Objekten in Produktion und Logistik*. Aachen: Apprimus-Verl, 2013.
- [21] T. Schoch and M. Strassner, "Wie smarte dinge prozesse unterstützen," in *Ubiquitous Computing* (H. Sauerburger, ed.), no. 229. Jg. 40 in HMD, pp. 23–32, Heidelberg: dpunkt-Verl., 2003.
- [22] G. Schuh and V. Stich, eds., *Logistikmanagement*. Springer Berlin Heidelberg, 2013.
- [23] G. G. Meyer, K. Främling, and J. Holmström, "Intelligent products: A survey," *Computers in Industry*, vol. 60, pp. 137–148, Apr. 2009.
- [24] P. J. S. II, *RFID For Dummies*. 2005.
- [25] L. Andrade, J. Figueiredo, and M. Tlemçani, "A new rfid-identification strategy applied to the marble extraction industry," *Electronics (Switzerland)*, vol. 10, no. 4, pp. 1–16, 2021.
- [26] B. J. Collins, "RFID Lands at Frankfurt Airport RFID Lands at Frankfurt Airport," *RFID Journal*, no. November 2003, pp. 1–5, 2005.
- [27] B. Mary and C. O. Connor, "LEGO Puts the RFID Pieces Together," *RFID Journal*, pp. 1–6, 2006.
- [28] Y. Álvarez López, J. Franssen, G. Álvarez Narciani, J. Pagnozzi, I. González-Pinto Arrillaga, and F. Las-Heras Andrés, "RFID technology for management and tracking: E-health applications," *Sensors (Switzerland)*, vol. 18, no. 8, pp. 1–17, 2018.
- [29] B. J. Collins, "SYSCO Gets Fresh With RFID," *RFID Journal*, pp. 1–5, 2005.
-

-
- [30] S. Chain, “Military Orders RFID Tracking Military Orders RFID Tracking,” *RFID Journal*, pp. 1–4, 2002.
- [31] V. D. Hunt, A. Puglia, and M. Puglia, *RFID-A Guide to Radio Frequency Identification*. 2006.
- [32] P. Sanghera, *RFID+ Study Guide and Practice Exams*. Syngress, 2007.
- [33] “Leitor Fixo em Portas - Imagem.” <https://www.codeway.biz/grav-en/user/pages/03.scanning/07.rfid-scanners/02.zebra-rfid-scanners/01.zebra-fx9500/fx9500-app2%20720x432.jpg>. Accessed: 01-05-2021.
- [34] “Leitor Portátil- Imagem.” <https://3.imimg.com/data3/HT/VU/MY-7047655/rfid-supply-chain-management-system-250x250.jpg>. Accessed: 01-05-2021.
- [35] “Leitor Empilhadeira- Imagem.” https://fr.hopelandrfid.com/js/htmledit/kindeditor/attached/20170929/20170929084958_69722.png. Accessed: 01-05-2021.
- [36] L. Kumari, K. Narsaiah, M. Grewal, and R. Anurag, “Application of RFID in agri-food sector,” *Trends in Food Science & Technology*, vol. 43, pp. 144–161, June 2015.
- [37] K. Domdouzis, B. Kumar, and C. Anumba, “Radio-Frequency Identification (RFID) applications: A brief introduction,” *Advanced Engineering Informatics*, vol. 21, no. 4, pp. 350–355, 2007.
- [38] D. Tanner, *Applications for RFID Technologies in the Food Supply Chain*. Elsevier, 2015.
- [39] “EPC - electronic product code (epc) - syntax.” <https://www.gs1.org/standards/epc-rfid>. Accessed: 01-05-2021.
- [40] “Tag Passivas - Imagem.” <http://www.xyrfid.com/wp-content/uploads/2017/08/hot-sell-uhf-rfid-iso180006c-printable-epc-5.jpg>. Accessed: 01-05-2021.
- [41] “Tag Ativa- Imagem.” <https://www.researchgate.net/publication/339643122/figure/fig3/AS:863357355696134@1582851842907/A-typical-semi-passive-tag-with-an-internal-battery-116.jpg>. Accessed: 01-05-2021.
- [42] “Smart Label - Imagem.” https://img.foodlogistics.com/files/base/acbm/fl/image/2017/11/1x1/Seagull_Scientific_smart_labels.5a159a5af3713.png. Accessed: 01-05-2021.
- [43] A. G. Costa, “Rochas ornamentais - recursos minerais de minas gerais - rmmg.” <http://recursomineralmg.codemge.com.br/substancias-minerais/rochas-ornamentais/>. Accessed: 01-05-2021.
-

-
- [44] C. Barros, S., Rosa, L., Pulino, P., Leitão, C., Agostinho, P., Leite, S., Bidarra, M., Botelho, J., Luís, “Produção de Rochas Ornamentais. Análise sectorial Banco BES,” tech. rep., Banco Espírito Santo, 2014.
- [45] A. Silva, *Tecnologias e práticas lean thinking na fileira das Rochas Ornamentais*. PhD thesis, Instituto Politécnico de Portalegre, 2014.
- [46] C. L. Luís Martins, Cristina Lourenço, J. F. d. C. José Alcântara Cruz, Patrícia Falé, and M. C. e. J. M. Martins, “Boletim de Minas,” tech. rep., Direção-Geral de Energia e Geologia, Lisboa, 2020.
- [47] C. G. P. S. Heberson Gomes Mendes, Mayke Gomes da Silva, Joel Costalonga Junior, “IMPACTOS AMBIENTAIS NO SETOR DE ROCHAS ORNAMENTAIS NO ESPÍRITO SANTO E ALOCAÇÃO E REAPROVEITAMENTO DE SEUS RESÍDUOS,” 2018.
- [48] D. Kaliampakos, A. Moutsatsou, and J. Skotaras, “Evaluation of wastes from marble quarries as paint fillers,” vol. 105, pp. C37–C42, 01 1996.
- [49] “EUROSTAT - statistics explained. european commission.” https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Waste_statistics. Accessed: 01-05-2021.
- [50] N. F. VIDAL, F. W. H., AZEVEDO, H. C. A., CASTRO, “Tecnologia de rochas ornamentais: pesquisa, lavra e beneficiamento,” tech. rep., CETEM – CENTRO DE TECNOLOGIA MINERAL FERNANDO, Rio de Janeiro, 2013.
- [51] D. M. H. Loke, “2-D and 3-D Electrical Imaging Surveys,” *Tutorial*, no. May, pp. 51–52, 2015.
- [52] M. H. Loke and R. D. Barker, “Practical techniques for 3D resistivity surveys and data inversion 1,” *Geophysical Prospecting*, vol. 44, no. 3, pp. 499–523, 1996.
- [53] M. H. Loke and T. Dahlin, “A comparison of the Gauss-Newton and quasi-Newton methods in resistivity imaging inversion,” *Journal of Applied Geophysics*, vol. 49, no. 3, pp. 149–162, 2002.
- [54] D. M. Loke, “Electrical imaging surveys for environmental and engineering studies,” *Cangkat Minden Lorong*, vol. 6574525, no. 1999, p. 63, 1999.
- [55] G.-M. Schwab, “Physical chemistry, von f. daniels und r. a. alberty. john wiley and sons, inc. new york. 1955. 1. aufl. VIII, 671 s., gebd. \$ 6.50,” *Angewandte Chemie*, vol. 68, pp. 252–252, Apr. 1956.
- [56] W. M. Telford, L. P. Geldart, and R. E. Sheriff, *Applied Geophysics*. Cambridge University Press, Oct. 1990.
- [57] G. V. Keller and F. C. Frischknecht, “Electrical methods in geophysical prospecting.” Pergamon Press Inc., 1966.

-
- [58] G. Software, “RES2DINVx64 features.” <http://www.geotomosoft.com/>, 2014. Accessed: 01-05-2021.
- [59] M. Mamillan, “”connaissances actuelles pour mesurer le degré d’altération des pierres et l’efficacité des méthodes de traitement”,” *1st International Sympo-Soium on the Deterioration of Building Stones*, p. 47–56, 1972.
- [60] M. Mamillan, “”méthodes d’essais physiques pour evaluer l’altération des pierres des monuments, the conservation of stone i”,” *The conservation of stone. I : proceedings / of the International symposium, Bologna, June 19-21, 1975*, pp. 595–634, 1975.
- [61] D. Deere and R. P. Miller, “Engineering classification and index properties for intact rock,” 1966.
- [62] Y. Youash, “Dynamic physical properties of rocks: Part ii, experimental results,” 1970.
- [63] V. S. Vutukuri, *Handbook on mechanical properties of rocks : testing techniques and results*. Clausthal Bay Village, Ohio Bay Village, Ohio: Trans Tech Publications Distributed by Trans Tech Publications, 1974.
- [64] *Relation Between Uniaxial Compressive Strength And Elastic Wave Velocity of Soft Rock*, vol. All Days of *ISRM International Symposium*, 09 1981. ISRM-IS-1981-002.
- [65] P. Gaviglio, “Longitudinal waves propagation in a limestone: The relationship between velocity and density,” *Rock Mechanics and Rock Engineering*, vol. 22, pp. 299–306, Oct. 1989.
- [66] E. Yasar and Y. Erdogan, “Correlating sound velocity with the density, compressive strength and young's modulus of carbonate rocks,” *International Journal of Rock Mechanics and Mining Sciences*, vol. 41, no. 5, pp. 871–875, 2004.
- [67] S. Kahraman, U. Ulker, and M. S. Delibalta, “A quality classification of building stones from P-wave velocity and its application to stone cutting with gang saws,” *Journal of the Southern African Institute of Mining and Metallurgy*, vol. 107, no. 7, pp. 427–430, 2007.
- [68] S. Kahraman, “The correlations between the saturated and dry p-wave velocity of rocks,” *Ultrasonics*, vol. 46, pp. 341–348, Nov. 2007.
- [69] A. Rahmouni, A. Boulanouar, M. Boukalouch, Y. Géraud, A. Samaouali, M. Harnafi, and J. Sebbani, “Prediction of porosity and density of calcarenite rocks from p-wave velocity measurements,” *International Journal of Geosciences*, vol. 04, no. 09, pp. 1292–1299, 2013.
- [70] F. Birch, “The velocity of compressional waves in rocks to 10 kilobars: 1.,” *Journal of Geophysical Research*, vol. 65, pp. 1083–1102, Apr. 1960.
- [71] S. Yagiz, “P-wave velocity test for assessment of geotechnical properties of some rock materials,” *Bulletin of Materials Science*, vol. 34, pp. 947–953, July 2011.
-

-
- [72] N. A. Al-Shayea, "Effects of testing methods and conditions on the elastic properties of limestone rock," *Engineering Geology*, vol. 74, pp. 139–156, July 2004.
- [73] M. Ciccotti and F. Mulargia, "Differences between static and dynamic elastic moduli of a typical seismogenic rock," *Geophysical Journal International*, vol. 157, pp. 474–477, Apr. 2004.
- [74] E. Eissa and A. Kazi, "Relation between static and dynamic young's moduli of rocks," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, vol. 25, pp. 479–482, Dec. 1988.
- [75] M. S. Ameen, B. G. Smart, J. Somerville, S. Hammilton, and N. A. Naji, "Predicting rock mechanical properties of carbonates from wireline logs (a case study: Arab-d reservoir, ghawar field, saudi arabia)," *Marine and Petroleum Geology*, vol. 26, pp. 430–444, Apr. 2009.
- [76] B. Christaras, F. Auger, and E. Mosse, "Determination of the moduli of elasticity of rocks. comparison of the ultrasonic velocity and mechanical resonance frequency methods with direct static methods," *Materials and Structures*, vol. 27, pp. 222–228, May 1994.
- [77] J. M. Ide, "Comparison of statically and dynamically determined young's modulus of rocks," *Proceedings of the National Academy of Sciences*, vol. 22, pp. 81–92, Feb. 1936.
- [78] M. King, "Static and dynamic elastic properties of rocks from the canadian shield," *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, vol. 20, pp. 237–241, Oct. 1983.
- [79] A. R. Najibi, M. Ghafoori, G. R. Lashkaripour, and M. R. Asef, "Empirical relations between strength and static and dynamic elastic properties of asmari and sarvak limestones, two main oil reservoirs in iran," *Journal of Petroleum Science and Engineering*, vol. 126, pp. 78–82, Feb. 2015.
- [80] M. Khandelwal and T. Singh, "Correlating static properties of coal measures rocks with p-wave velocity," *International Journal of Coal Geology*, vol. 79, pp. 55–60, July 2009.
- [81] T. Acharya, *Image processing : principles and applications*. Hoboken, N.J: John Wiley, 2005.
- [82] B. K. Behera, "IMAGE-PROCESSING IN TEXTILES," *Textile Progress*, vol. 35, pp. 1–193, June 2004.
- [83] L. Cutaiia, P. Massacci, and I. Roselli, "Analysis of landsat 5 TM images for monitoring the state of restoration of abandoned quarries," *International Journal of Surface Mining, Reclamation and Environment*, vol. 18, pp. 122–134, June 2004.
-

-
- [84] D. KARAKUŞ, *DETERMINATION OF THE STRUCTURAL PROPERTIES OF ROCKS WITH IMAGE ANALYSING METHODS*. PhD thesis, DOKUZ EYLÜL ÜNİVERSİTESİ, 2006.
- [85] G. Lane, C. Martin, and E. Pirard, “Techniques and applications for predictive metallurgy and ore characterization using optical image analysis,” *Minerals Engineering*, vol. 21, pp. 568–577, June 2008.
- [86] N. A. Baykan and N. Yılmaz, “Mineral identification using color spaces and artificial neural networks,” *Computers & Geosciences*, vol. 36, pp. 91–97, Jan. 2010.
- [87] N. Maerz, “Aggregate sizing and shape determination using digital image processing,” 1998.
- [88] E. Cabello, M. Sánchez, and J. Delgado, “A new approach to identify big rocks with applications to the mining industry,” *Real-Time Imaging*, vol. 8, pp. 1–9, Feb. 2002.
- [89] N. Maerz and Palangio, “Online fragmentation analysis for grinding and crushing control,” 2000.
- [90] M. Bailey, C. Gomez, and J. Finch, “Development and application of an image analysis method for wide bubble size distributions,” *Minerals Engineering*, vol. 18, pp. 1214–1221, Oct. 2005.
- [91] J. Liu, J. MacGregor, C. Duchesne, and G. Bartolacci, “Flotation froth monitoring using multiresolutional multivariate image analysis,” *Minerals Engineering*, vol. 18, pp. 65–76, Jan. 2005.
- [92] Zafir EKMEKÇİ;Ayşe Nihan, “Analysis of the Relationship between Froth Appearance and Flotation Performance by Machine Vision System,” vol. 45, pp. 27–38, 2006.
- [93] O. Akkoyun, “An evaluation of image processing methods applied to marble quality classification,” in *2010 2nd International Conference on Computer Technology and Development*, IEEE, Nov. 2010.
- [94] E. Ardalı, “Dokuz eylul university graduate school of natural and applied sciences classification of marble textures using neural networks and image processing methods,” 2008.
- [95] T. Seemann, *Digital Image Processing using Local Segmentation*. PhD thesis, Faculty of Information Technology Monash University, 2002.
- [96] J. M. C. Sousa and J. R. C. Pinto, “Comparison of intelligent classification techniques applied to marble classification,” in *Lecture Notes in Computer Science*, pp. 802–809, Springer Berlin Heidelberg, 2004.
- [97] J. Miguel and P. Batista, “Marble Polished Stones Automatic Classification,” pp. 1–8, 2015.

-
- [98] J. M.-C. de Vaca-Alajarín and L. Tomás-Balibrea, “Marble slabs quality classification system using texture recognition and neural networks methodology,” in *ESANN*, 1999.
- [99] M. Deviren, M. K. Balci, U. M. Leloglu, and M. Severcan, “A feature extraction method for marble tile classification,” 2000.
- [100] M. Gökay and I. Gundogdu, “Color identification of some turkish marbles,” *Construction and Building Materials*, vol. 22, pp. 1342–1349, July 2008.
- [101] J. Martinez-Alajarin, J. Luis-Delgado, and L. Tomas-Balibrea, “Automatic system for quality-based classification of marble textures,” *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 35, pp. 488–497, Nov. 2005.
- [102] S. Priest, *Discontinuity Analysis for Rock Engineering*. Dordrecht: Springer Netherlands Imprint Springer, 1993.
- [103] R. Nelson, *Geologic analysis of naturally fractured reservoirs*. Boston: Gulf Professional Pub, 2001.
- [104] Z. T. Bieniawski, *Engineering rock mass classifications : a complete manual for engineers and geologists in mining, civil, and petroleum engineering*. New York: Wiley, 1989.
- [105] R. Roncella, G. Forlani, and F. Remondino, “Photogrammetry for geological applications: automatic retrieval of discontinuity orientation in rock slopes,” in *Videometrics VIII* (J.-A. Beraldin, S. F. El-Hakim, A. Gruen, and J. S. Walton, eds.), SPIE, Jan. 2005.
- [106] G. Gigli and N. Casagli, “Semi-automatic extraction of rock mass structural data from high resolution LIDAR point clouds,” *International Journal of Rock Mechanics and Mining Sciences*, vol. 48, pp. 187–198, Feb. 2011.
- [107] J. Otoo, N. Maerz, Y. Duan, and L. Xiaoling, “Lidar and optical imaging for 3-d fracture orientations,” 2011.
- [108] M. Pötsch, W. Schubert, and A. Gaich, “Application of metric 3d images of rock faces for the determination of the response of rock slopes to excavation,” in *Impact of Human Activity on the Geological Environment* (P. Konecný, ed.), (United Kingdom), pp. 489–497, 2005.
- [109] S. Slob, H. Hack, Q. Feng, K. Röshoff, and A. Turner, “Fracture mapping using 3D laser scanning techniques,” *11th Congress of the International Society for Rock Mechanics*, no. July, pp. pp. 299–302., 2007.
- [110] M. Works, “Matlab reference guide.” <https://www.mathworks.com/help/matlab/>, 2021. Accessed: 01-05-2021.
- [111] S. AG, “RFU620-10100 DATASHEET.” https://cdn.sick.com/media/pdf/2/52/052/dataSheet_RFU620-10100_1062599_en.pdf. Accessed: 01-05-2021.
-

-
- [112] S. AG, "SOPAS ET 2018.4 EN." <https://www.sick.com/ag/en/sopas-engineering-tool-2018/p/p367244>. Accessed: 01-05-2021.
- [113] S. AG, "Simatic s7-1200 cpu 1214c. 6es7 214-1bg40-0xb0 datasheet." http://www.tic.co.th/file_upload/download/20180308095829.pdf. Accessed: 01-05-2021.
- [114] S. AG, "Simatic hmi ktp 600 basic color pn. 6av6 647-0ad1-3ax0 datasheet." <https://datasheet.octopart.com/6AV6647-0AD11-3AX0-Siemens-datasheet-12174366.pdf>. Accessed: 01-05-2021.
- [115] S. AG, "Simatic hmi wincc. wincc runtime advanced readme." https://cache.industry.siemens.com/dl/files/803/109753803/att_937533/v1/ReadMe_WinCC_RT_Advanced_V15_enUS.pdf. Accessed: 01-05-2021.
- [116] S. AG, "Delivery release tia portal v15." <https://support.industry.siemens.com/cs/document/109752224/delivery-release-tia-portal-v15?dti=0&lc=en-WW>. Accessed: 01-05-2021.
- [117] "SQL Server manual reference." <https://docs.microsoft.com/en-gb/sql/?view=sql-server-ver15>. Accessed: 01-05-2021.
- [118] "RAD Studio 10.4.2." <https://www.embarcadero.com/products/rad-studio?aldSet=en-GB>. Accessed: 01-05-2021.
- [119] "Delphi - language overview." http://docwiki.embarcadero.com/RADStudio/Sydney/en/Language_Overview. Accessed: 01-05-2021.
- [120] "MySQL manual reference." <https://dev.mysql.com/doc/>. Accessed: 01-05-2021.
- [121] "HTML 5 reference - the syntax, vocabulary and apis of html5." <https://dev.w3.org/html5/html-author/>. Accessed: 01-05-2021.
- [122] "PHP Manual - language reference." <https://www.php.net/manual/en/langref.php>. Accessed: 01-05-2021.
- [123] "CSS - language reference." <https://cssreference.io/>. Accessed: 01-05-2021.
- [124] "Bootstrap - getting started." <https://getbootstrap.com/docs/4.4/getting-started/introduction/>. Accessed: 01-05-2021.
- [125] "SQL Reference Manual." http://trafodion.apache.org/docs/sql_reference/index.html. Accessed: 01-05-2021.
- [126] "HeidiSQL manual reference." <https://www.heidisql.com/help.php>. Accessed: 01-05-2021.
- [127] "VBScript Features." [https://docs.microsoft.com/en-us/previous-versions/273zc69c\(v=vs.85\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/273zc69c(v=vs.85)?redirectedfrom=MSDN). Accessed: 01-05-2021.
-

-
- [128] “Structured Control Language (SCL) for S7-300/S7-400 Programming.” https://cache.industry.siemens.com/d1/files/188/1137188/att_27471/v1/SCLV4_e.pdf. Accessed: 01-05-2021.

A ANEXOS

A.0.1 CODIGO DELPHI - RFID SYSTEM

```
1: unit RFID_CLIENT;
2:
3: interface
4:
5: uses
6: Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graph
   ics,
7: Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.ComCtrls, Vcl.ExtCtrls, Vcl.StdCtrls,
8: System.Win.ScktComp, FireDAC.Stan.Intf, FireDAC.Stan.Option,
9: FireDAC.Stan.Error, FireDAC.UI.Intf, FireDAC.Phys.Intf, FireDAC.Stan.Def,
10: FireDAC.Stan.Pool, FireDAC.Stan.Async, FireDAC.Phys, FireDAC.Phys.MySQL,
11: FireDAC.Phys.MySQLDef, FireDAC.VCLUI.Wait, Data.DB, Vcl.Grids, Vcl.DBGrids,
12: Datasnap.DBClient, FireDAC.Comp.Client, FireDAC.Stan.Param, FireDAC.DatS,
13: FireDAC.DApt.Intf, FireDAC.DApt, FireDAC.Comp.DataSet;
14:
15: type
16: TRFID_FORM = class (TForm)
17:   Panel1: TPanel;
18:   PageControl1: TPageControl;
19:   TabSheet1: TTabSheet;
20:   TabSheet2: TTabSheet;
21:   TabSheet3: TTabSheet;
22:   TabSheet4: TTabSheet;
23:   Label1: TLabel;
24:   Label2: TLabel;
25:   ip_edit: TEdit;
26:   port_edit: TEdit;
27:   Button1: TButton;
28:   Bdisconnect_sensor: TButton;
29:   database_edit: TEdit;
30:   host_edit: TEdit;
31:   pass_edit: TEdit;
32:   user_edit: TEdit;
33:   Label3: TLabel;
34:   Label4: TLabel;
35:   Label5: TLabel;
36:   Label6: TLabel;
37:   Bconnect_database: TButton;
38:   Bdisconnect_database: TButton;
39:   terminal: TMemo;
40:   CONN_SENSOR: TClientSocket;
41:   Label7: TLabel;
42:   Label8: TLabel;
43:   Label9: TLabel;
44:   obj_code_edit: TEdit;
45:   obj_name_edit: TEdit;
46:   Button2: TButton;
47:   CONN_DATA: TFDConnection;
48:   MySQLDriverLink1: TFDPhysMySQLDriverLink;
49:   DBGrid1: TDBGrid;
50:   Label10: TLabel;
51:   SELECT_OBJECT: TFDQuery;
52:   DataSource1: TDataSource;
53:   Timer1: TTimer;
54:   SELECT_SCANS: TFDQuery;
55:   Label11: TLabel;
56:   AREA_EDIT: TEdit;
57:   procedure Button1Click(Sender: TObject);
58:   procedure CONN_SENSORRead(Sender: TObject; Socket: TCustomWinSocket);
59:   procedure Bdisconnect_sensorClick(Sender: TObject);
60:   procedure Bconnect_databaseClick(Sender: TObject);
61:   procedure Bdisconnect_databaseClick(Sender: TObject);
62:   procedure Timer1Timer(Sender: TObject);
63:   procedure Button2Click(Sender: TObject);
64:
65:
66:
67:
68:
69:   private
70:     { Private declarations }
71:   public
72:     { Public declarations }
73: end;
```

```

74:
75: var
76:   RFID_FORM: TRFID_FORM;
77:
78: implementation
79:
80: {$R *.dfm}
81:
82: procedure TRFID_FORM.Bconnect_databaseClick(Sender: TObject);
83: begin
84:
85: CONN_DATA.Connected := false;
86: CONN_DATA.DriverName := 'MySQL';
87: CONN_DATA.Params.Add('Server='+HOST_EDIT.Text);
88: CONN_DATA.Params.Add('Database='+DATABASE_EDIT.Text);
89: CONN_DATA.Params.Add('User_name='+USER_EDIT.TEXT);
90: CONN_DATA.Params.Add('Password='+Pass_EDIT.TEXT);
91: CONN_DATA.Connected := True;
92:
93: end;
94:
95: procedure TRFID_FORM.Bdisconnect_databaseClick(Sender: TObject);
96: begin
97: CONN_DATA.Connected:=False;
98: end;
99:
100: procedure TRFID_FORM.Bdisconnect_sensorClick(Sender: TObject);
101:
102:
103: begin
104: CONN_SENSOR.Active:=False;
105:
106: end;
107:
108: procedure TRFID_FORM.Button1Click(Sender: TObject);
109:
110:
111: begin
112: CONN_SENSOR.Active:=False;
113: CONN_SENSOR.Port:=strtoint(PORT_EDIT.Text);
114: CONN_SENSOR.Address:=IP_EDIT.Text;
115: CONN_SENSOR.Active:=True;
116:
117: end;
118:
119: procedure TRFID_FORM.Button2Click(Sender: TObject);
120: var
121:
122: object_code:string;
123: object_name:string;
124:
125: begin
126:   object_code:=#39+trim(obj_code_edit.Text)+#39;
127:   object_name:=#39+trim(obj_name_edit.Text)+#39;
128:   if CONN_DATA.Connected then
129:     begin
130:       CONN_DATA.ExecSQL( 'INSERT INTO andrade9653_work.TABLE_OBJECTS VALUES('+object_code+', '+
131:         object_name+')');
132:     end
133: end;
134:
135: procedure TRFID_FORM.CONN_SENSORRead(Sender: TObject; Socket: TCustomWinSocket);
136: var
137: dados:string;
138: today : TDateTime;
139: mydate:string;
140: mytime:string;
141: scan_data :string;
142: begin
143: today:=now;
144: mydate:= #39+formatdatetime('yyyy/mm/dd', today)+#39;
145: mytime:=#39+formatdatetime('HH:MM:SS', today)+#39;
146:

```



```
147: dados:=CONN_SENSOR.Socket.ReceiveText;
148: terminal.Text:=terminal.Text+trim(dados)+#13#10;
149:
150:   if CONN_DATA.Connected then
151:     begin
152:
153:       SELECT_SCANS.Active:=True;
154:       scan_data:=SELECT_SCANS.Fields[2].AsString;
155:
156:
157:       if not (dados.Equals(scan_data)) then
158:         begin
159:           dados:=#39+trim(dados)+#39;
160:           CONN_DATA.ExecSQL( 'INSERT INTO andrade9653_work.TABLE_SCANS VALUES('+mytime+', '+m
161: ydate+', '+dados+', '+ trim(area_edit.Text)+')');
162:         end;
163:
164:
165:       end
166:
167: end;
168:
169: procedure TRFID_FORM.Timer1Timer(Sender: TObject);
170: begin
171:
172:   if CONN_DATA.Connected then
173:     begin
174:       SELECT_OBJECT.Active:=False;
175:       SELECT_OBJECT.Active:=True;
176:
177:     end
178:
179:
180: end;
181:
182: end.
```

A.0.2 CODIGO DELPHI - DATATRANSFER

```
1: unit TRANSFER_DATA;
2:
3: interface
4:
5: uses
6:   Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes, Vcl.Graph
   ics,
7:   Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.ComCtrls, Vcl.ExtCtrls,
8:   FireDAC.Phys.MySQLDef, FireDAC.Stan.Intf, FireDAC.Stan.Option,
9:   FireDAC.Stan.Error, FireDAC.UI.Intf, FireDAC.Phys.Intf, FireDAC.Stan.Def,
10:  FireDAC.Stan.Pool, FireDAC.Stan.Async, FireDAC.Phys, FireDAC.Phys.MySQL,
11:  FireDAC.VCLUI.Wait, FireDAC.Phys.MSSQLDef, FireDAC.Phys.MSSQL,
12:  FireDAC.Phys.ODBCBase, Data.DB, FireDAC.Comp.Client, FireDAC.Stan.Param,
13:  FireDAC.DatS, FireDAC.DApt.Intf, FireDAC.DApt, Vcl.Grids, Vcl.DBGrids,
14:  FireDAC.Comp.DataSet;
15:
16: type
17:   TForm1 = class(TForm)
18:     Panel1: TPanel;
19:     FDPPhysMySQLDriverLink1: TFDPhysMySQLDriverLink;
20:     CONN_MYSQL: TFDConnection;
21:     FDPPhysMSSQLDriverLink1: TFDPhysMSSQLDriverLink;
22:     CONN_SSQL: TFDConnection;
23:     SELECT_MYSQL: TFDQuery;
24:     SELECT_SSQL: TFDQuery;
25:     DataSource1: TDataSource;
26:     DataSource2: TDataSource;
27:     PageControl1: TPageControl;
28:     TabSheet1: TTabSheet;
29:     Panel3: TPanel;
30:     TABLE_MYSQL: TDBGrid;
31:     Panel2: TPanel;
32:     TABLE_SSQL: TDBGrid;
33:     Panel4: TPanel;
34:     TabSheet2: TTabSheet;
35:     Label1: TLabel;
36:     Label2: TLabel;
37:     Label3: TLabel;
38:     Label4: TLabel;
39:     bDISCONN_MYSQL: TButton;
40:     Button1: TButton;
41:     HOST_MYSQL_EDIT: TEdit;
42:     USER_MYSQL_EDIT: TEdit;
43:     PASS_MYSQL_EDIT: TEdit;
44:     DATANAME_MYSQL_EDIT: TEdit;
45:     TabSheet3: TTabSheet;
46:     Label5: TLabel;
47:     Label6: TLabel;
48:     Label7: TLabel;
49:     Label8: TLabel;
50:     bCONN_SQL: TButton;
51:     bDISCONN_SQL: TButton;
52:     PASS_SQL_EDIT: TEdit;
53:     DATANAME_SQL_EDIT: TEdit;
54:     USER_SQL_EDIT: TEdit;
55:     HOST_SQL_EDIT: TEdit;
56:     Button2: TButton;
57:     Button3: TButton;
58:     Timer1: TTimer;
59:     SELECT_OBJ_SSQL: TFDQuery;
60:     SELECT_OBJ_MYSQL: TFDQuery;
61:     area_edit: TEdit;
62:     Label9: TLabel;
63:     procedure Button2Click(Sender: TObject);
64:     procedure Timer1Timer(Sender: TObject);
65:     procedure Button3Click(Sender: TObject);
66:
67:   private
68:     { Private declarations }
69:   public
70:     { Public declarations }
71:   end;
72:
73: var
```

```

74:   Form1: TForm1;
75:   flag_operation:boolean =False;
76: implementation
77:
78: {$R *.dfm}
79:
80:
81:
82: procedure TForm1.Button2Click(Sender: TObject);
83:
84:
85: begin
86:
87:   flag_operation:=True;
88:
89: end;
90:
91: procedure TForm1.Button3Click(Sender: TObject);
92: begin
93:   flag_operation:=False;
94: end;
95:
96: procedure TForm1.Timer1Timer(Sender: TObject);
97: var
98:   mysql_line:string;
99:   ssl_line:string;
100:  dados:string;
101:  mytime:string;
102:  mydate:string;
103:  temp :String;
104:  today:TDATEIME;
105:  test:string;
106:  area_number:integer;
107: begin
108:   today:=now;
109:   mydate:=#39+formatdatetime('yyyy/mm/dd',today)+#39;
110:   mytime:=#39+formatdatetime('HH:MM:SS',today)+#39;
111:
112:
113:
114:   if flag_operation then
115:     begin
116:
117:
118:       if CONN_MYSQL.Connected then
119:         begin
120:           SELECT_MYSQL.Active:=False;
121:           SELECT_MYSQL.Active:=True;
122:           SELECT_OBJ_MYSQL.Open;
123:
124:           temp := SELECT_OBJ_MYSQL.FieldByName('OBJECT_NAME').AsString;
125:
126:
127:         end;
128:       if CONN_SSQL.Connected then
129:         begin
130:           SELECT_SSQL.Active:=False;
131:           SELECT_SSQL.Active:=True;
132:
133:         end;
134:
135:
136:       mysql_line:=TABLE_MYSQL.DataSource.DataSet.FieldByName('OBJECT_NAME').AsString;
137:       ssl_line:=TABLE_SSQL.DataSource.DataSet.FieldByName('NAME').AsString;
138:       area_number:=strtoint(trim(area_edit.Text));
139:
140:       if mysql_line.Equals(ssl_line) then
141:         begin
142:           //ShowMessage('Lines are Equals'+ssl_line);
143:         end
144:       else
145:         begin
146:           //ShowMessage('Lines are Differents');
147:           dados:=#39+ssl_line+#39;

```

```
148:         CONN_MYSQL.ExecSQL( 'INSERT INTO andrade9653_work.TABLE_SCANS VALUES('+mytime+', '+my
149:         date+', '+dados+', '+inttostr(area_number) +')');
150:             end;
151:         end;
152:     end;
153:
154: end.
```

A.0.3 PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO TRCV_C

TRCV_C: Receive data via Ethernet (S7-1200)

TRCV_C: Receive data via Ethernet



Validity

The following description of the "TRCV_C" instruction applies to S7-1200 CPUs with firmware version < V4.0.

Description

The "TRCV_C" instruction is executed asynchronously and has the following functions:

1. Setting up and establishing a communication connection:

"TRCV_C" sets up and establishes a TCP or ISO-on-TCP communication connection. Once the connection has been set up and established, it is automatically maintained and monitored by the CPU.

The connection description specified at the CONNECT parameter is used to set up the communication connection. To establish a connection, the CONT parameter must be set to the value "1". Once the connection is successfully established, the DONE parameter is set to "1".

An existing connection is terminated and the connection which has been set up is removed when the CPU goes into STOP mode. To set up and establish the connection again, you must execute "TRCV_C" again.

For information on the number of possible communication connections, refer to the technical specifications for your CPU.

2. Receiving data via an existing communication connection:

If the EN_R parameter is set to the value "1", receipt of data is enabled. When receiving data (rising edge at the EN_R parameter), the CONT parameter must have the value TRUE in order to establish or maintain a connection.

The received data is entered in a receive area. You specify the length of the receive area either with the LEN parameter (if LEN <> 0) or with the length information of the DATA parameter (if LEN = 0) in accordance with the protocol variant used. If you use purely symbolic values at the DATA parameter, the LEN parameter must have the value "0".

After data has been received successfully, the signal state at the DONE parameter is "1". If errors occur in the data transfer, the DONE parameter is set to "0".

3. Terminating the communication connection:

The communication connection is terminated immediately when the CONT parameter is set to "0".

TRCV_C is executed again when the COM_RST parameter is set. This terminates the existing communication connection and a new connection is established. If data is being received when it executes again, this can lead to a loss of data.

Note

Support when programming connections

If you select an instruction for communication TCON, TSEND_C or TRCV_C in a program block and create connections of the type TCP, UDP or ISO-on-TCP and want to assign parameters to them, you can use the support of the connection parameter assignment.

You can find the connection parameter assignment in the Inspector window of the program editor.

Receive modes of TRCV_C

The following table shows how the received data is entered in the receive area.

Protocol variant	Availability of data in the receive area	connection_type parameter of the connection description	LEN parameter
TCP (Ad-hoc mode)	The data is immediately available.	Hexadecimal value: B#16#11 Integer value: 17	0
TCP (Receipt of data with specified length)	The data is available as soon as the data length specified at the LEN parameter has been fully received.	Hexadecimal value: B#16#11 Integer value: 17	1 to 8192
ISO on TCP (Message-oriented data transfer)	The data is available as soon as the data length specified at the LEN parameter has been fully received.	Hexadecimal value: B#16#12 Integer value: 18	<ul style="list-style-type: none"> • 1 to 1452, if a CP is used. • 1 to 8192, if no CP is used.

TCP (Ad-hoc mode)

The ad-hoc mode is only available with the TCP protocol variant. You use the ad-hoc mode to receive data with dynamic length with the "TRCV" instruction.

You set ad-hoc mode by assigning the value "0" to the LEN parameter. All data types can be used for data blocks with standard access when you use ad-hoc mode. Only ARRAY of BYTE or data types with a length of 8 bits can be used for data blocks with optimized access (e.g., CHAR, USINT, SINT, etc.). The data length actually received is output at the RCVD_LEN parameter.

TCP (Receipt of data with specified length)

You use the value of the LEN parameter to specify the length for the data receipt. The data receipt is not complete until the length of data specified at the LEN parameter has been completely received. Only then is the data available in the receive area (DATA parameter). The actually received data length in bytes at the RCVD_LEN parameter corresponds to the data length at the LEN parameter after receipt.

ISO on TCP (Message-oriented data transfer)

Complete message blocks are sent via a connection with the protocol variant ISO on TCP; these are recognized as such by the recipient. When using ISO on TCP, "TRCV_C" signals data receipt as soon as the message block has been completely received. The receive area is defined by the LEN and DATA parameters. If the receive buffer (DATA parameter) is too small for the sent data, "TRCV_C" signals an error. The actually received data length in bytes at the RCVD_LEN parameter corresponds to the data length at the LEN parameter after receipt.

Parameters

The following table shows the parameters of the "TRCV_C" instruction:

Parameter	Declaration	Data type	Memory area	Description
EN_R	Input	BOOL	I, Q, M, D, L	Receive enable
CONT	Input	BOOL	I, Q, M, D, L	Controls the communication connection: <ul style="list-style-type: none"> • 0: Disconnect the communication connection • 1: Establish and maintain the communication connection When receiving data (rising edge at the EN_R parameter), the CONT parameter must have the value TRUE in order to establish or maintain a connection.
LEN	Input	UINT	I, Q, M, D, L or constant	Maximum length of the data to be received (maximum value: 8192 bytes). If you use purely symbolic values at the DATA parameter, the LEN parameter must have the value "0".
CONNECT	InOut	TCON_Param	D	Pointer to the connection description See also: Connection parameters with structure according to TCON_Param
DATA	InOut	VARIANT	I, Q, M, D, L	Pointer to the receive area When transferring structures, the structures must be identical at the sending and receiving end.
COM_RST	InOut	BOOL	I, Q, M, D, L	Restarts the instruction: <ul style="list-style-type: none"> • 0: Irrelevant • 1: Complete restart of the instruction causing an existing connection to be terminated
DONE	Output	BOOL	I, Q, M, D, L	Status parameter with the following values: <ul style="list-style-type: none"> • 0: Job not yet started or is still executing • 1: Job executed without errors
BUSY	Output	BOOL	I, Q, M, D, L	Status parameter with the following values: <ul style="list-style-type: none"> • 0: Job not yet started or already completed • 1: Job not yet completed. A new job cannot be started
ERROR	Output	BOOL	I, Q, M, D, L	Status parameter ERROR: <ul style="list-style-type: none"> • 0: No error • 1: Error occurred

STATUS	Output	WORD	I, Q, M, D, L	Status of the instruction
RCVD_LEN	Output	UINT	I, Q, M, D, L	Amount of data actually received in bytes

You can find additional information on valid data types under "[Overview of the valid data types](#)".

BUSY, DONE and ERROR parameters

You can check the status of the job with the BUSY, DONE, ERROR and STATUS parameters. The BUSY parameter indicates the processing status. With the DONE parameter, you can check whether or not a job executed successfully. The ERROR parameter is set when errors occurred during execution of "TRCV_C". The error information is output at the STATUS parameter.

The following table shows the relationship between the BUSY, DONE and ERROR parameters:

BUSY	DONE	ERROR	Description
1	-	-	The job is being processed.
0	1	0	The job was completed successfully.
0	0	1	The job ended with an error. The cause of the error is output at the STATUS parameter.
0	0	0	No new job was assigned.

ERROR and STATUS parameters

ERROR	STATUS* (W#16#...)	Description
0	0000	Job completed without error.
0	0001	The connection establishment is complete.
0	0003	The connection termination is complete.
0	7000	No job processing active.
0	7001	<ul style="list-style-type: none"> • Start execution of the job • Establish connection • Wait for connection partner
0	7002	Data is being received.
0	7003	Connection is being terminated.
0	7004	<ul style="list-style-type: none"> • Connection established and monitored • No job processing active
0	7006	Data is currently being received.
1	8085	<ul style="list-style-type: none"> • The LEN parameter is larger than the highest permitted value. • The value at the LEN or DATA parameter was changed after the first call.
1	8086	The ID parameter is outside the permitted range.
1	8087	Maximum number of connections reached; no additional connection possible

1	8088	The value at the LEN parameter does not correspond to the receive area set at the DATA parameter.
1	8089	The CONNECT parameter does not point to a data block.
1	8091	Maximum nesting depth exceeded.
1	809A	The CONNECT parameter points to a field that does not correspond to the length of the connection description.
1	809B	The ID of the local device (local_device_id) in the connection description does not correspond to the CPU.
1	80A0	Group error for error codes W#16#80A1 and W#16#80A2.
1	80A1	<ul style="list-style-type: none"> • Connection or port already being used by user. • Communication error: <ul style="list-style-type: none"> ◦ The specified connection has not yet been established. ◦ The specified connection is being terminated. <p>Transfer via this connection is not possible.</p> <ul style="list-style-type: none"> ◦ The interface is being re-initialized.
1	80A2	Local or remote port is being used by the system.
1	80A3	<ul style="list-style-type: none"> • Attempt being made to re-establish an existing connection. • Attempt being made to terminate a non-existent connection.
1	80A4	IP address of the remote endpoint of the connection is invalid, in other words, it matches the IP address of the local partner.
1	80A7	Communication error: You called the instruction with COM_RST = 1 before the send job was complete.
1	80B2	The CONNECT parameter points to a data block that was generated with the attribute "Only store in load memory".
1	80B3	Inconsistent parameter assignment: Group error for error codes W#16#80A0 to W#16#80A2, W#16#80A4, W#16#80B4 to W#16#80B9.
1	80B4	When using the protocol variant ISO on TCP (connection_type = B#16#12) for the passive establishment of a connection (active_est = FALSE) one or both of the following conditions was violated: "local_tsap_id_len >= B#16#02" and/or "local_tsap_id[1] = B#16#E0".
1	80B5	Only passive connection establishment is permitted for connection type 13 = UDP.
1	80B6	Parameter assignment error in the connection_type parameter of the data block for connection description.
1	80B7	Error in one of the following parameters of the data block for connection description: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	80C3	<ul style="list-style-type: none"> • All connection resources are in use. • A block with this ID is already being processed in a different priority group.
1	80C4	<p>Temporary communication error:</p> <ul style="list-style-type: none"> • The connection cannot be established at this time. • The connection cannot be established because the firewalls on the connection path are not open for the required ports. • The interface is receiving new parameters or the connection is being established.

		<ul style="list-style-type: none"> • The configured connection is currently being removed by a "TDISCON" instruction. • The connection used is being terminated by a call with COM_RST=1.
1	80C6	The remote partner cannot be reached (network error).
1	8722	Error in the CONNECT parameter: Invalid source area (area not declared in data block).
1	873A	Error in the CONNECT parameter: Access to connection description is not possible (no access to data block).
1	877F	Error in the CONNECT parameter: Internal error
1	8922	Parameter DATA: Invalid destination area; the area does not exist in the DB.
1	8924	Parameter DATA: Area error in the VARIANT pointer.
1	8932	Parameter DATA: The DB number is too high.
1	893A	Parameter CONNECT: Access to specified connection data not possible (e.g. because the DB does not exist).
1	897F	Parameter DATA: Internal error, e.g. invalid VARIANT reference.
1	8A3A	Parameter DATA: No access to the data area, for example because the data block does not exist.
* The error codes can be displayed as integer or hexadecimal values in the program editor. For information on switching the display formats, refer to "See also".		

Note

Error messages of the instructions "TCON", "TRCV" and "TDISCON"

Internally, the TRV_C instruction uses the "**TCON**", "**TRCV**" and "**TDISCON**" instructions. The error messages of these instructions are contained in the respective descriptions.

See also

[Principle of operation of connection-oriented protocols \(S7-1200, S7-1500\)](#)
[Basics of Open User Communication \(S7-1200, S7-1500\)](#)
[Difference between synchronous and asynchronous instructions \(S7-1200, S7-1500\)](#)
[Overview of connection configuration \(S7-1200, S7-1500\)](#)
[Starting connection parameter assignment \(S7-1200, S7-1500\)](#)
[Creating and assigning parameters to connections \(S7-1200, S7-1500\)](#)

A.0.4 PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO FIND

FIND: Find characters in a character string (S7-1200, S7-1500)

FIND: Find characters in a character string



Description

You use this instruction to search through the character string at the IN1 input parameter for a specific character string.

- You specify the value to be searched for in the IN2 input parameter. The search is made from left to right.
- The position of the first occurrence is output in the OUT output parameter. If the search returns no match, the value "0" will be output in the OUT output parameter.

If an invalid character is specified at the IN2 parameter or if an error occurs during processing, the value "0" is output at the OUT parameter.

Parameters

The following table shows the parameters of the "FIND" instruction:

Parameter	Declaration	Data type	Memory area	Description
IN1	Input	STRING, WSTRING	D, L or constant	String searched through
IN2	Input	STRING, WSTRING	D, L or constant	Character string that is searched for
OUT	Return	INT	I, Q, M, D, L	Character position

You can find additional information on valid data types under "[Overview of the valid data types](#)".

Example

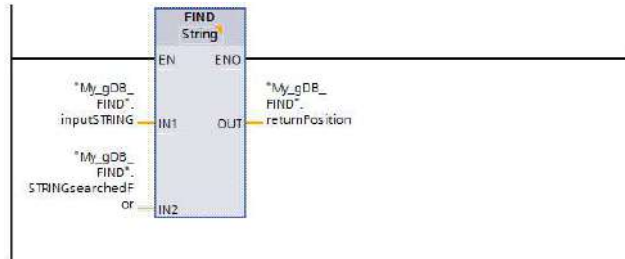
In the following example, you search for a character string in another character string. The data type used is STRING.

Create three tags in a global data block for storing the data.

My_gDB_FIND			
	Name	Data type	Start value
1	Static		
2	InputSTRING	String	'1234abcd'
3	STRINGsearchedFor	String	'4a'
4	returnPosition	Int	0

Interconnect the parameters of the instruction as follows.

FIND: Find characters in a character string (S7-1200, S7-1500)



The first character string ("inputSTRING") searches for the value "4a" of the second character string ("STRINGsearchedFor"). The character position ("4") at which the searched character string begins is output at output parameter OUT ("returnPosition").

My_gDB_FIND				
	Name	Data type	Start value	Monitor value
1	Static			
2	inputSTRING	String	'1234abcd'	'1234abcd'
3	STRINGsearchedFor	String	'4a'	'4a'
4	returnPosition	Int	0	4

You can find additional information and the program code for the above named example here: [Sample Library for Instructions](#)

A.0.5 PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO MID

MID: Read middle characters of a character string (S7-1200, S7-1500)

MID: Read middle characters of a character string



Description

You use this instruction to extract a portion of the character string in the IN input parameter. With the P parameter, you specify the position of the first character to be extracted. With the L parameter, you define the length of the character string to be extracted. The extracted partial character string is output to the OUT output parameter.

Note the following rules when executing the instruction:

- If the number of characters to be extracted exceeds the current length of the character string in the IN input parameter, a partial character string will be output. The partial character string starts at the P character string and is continued until the end of the character string.
- If the character position specified in the P parameter falls outside the current character string length in the IN input parameter, an empty character string will be output in the OUT output parameter.
- If the value of the P or L parameter equals zero or is negative, an empty character string will be output in the OUT output parameter.

If errors occur during processing of the instruction and it is possible to write the OUT output parameter, an empty string will be output.

Parameters

The following table shows the parameters of the "MID" instruction:

Parameter	Declaration	Data type	Memory area	Description
IN	Input	STRING, WSTRIN G	D, L or constant	Character string
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L or constant	Length of the string to be ex- tracted
P	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L or constant	Position of the first character to be extracted (first character = 1)
OUT	Return	STRING, WSTRIN G	D, L	Extracted partial string

You can find additional information on valid data types under "[Overview of the valid data types](#)".

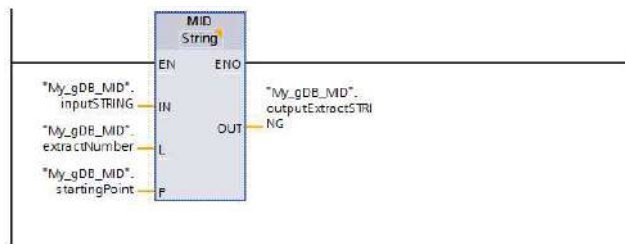
Example

In the following example, you extract a partial character string with the STRING data type from the middle of a character string.

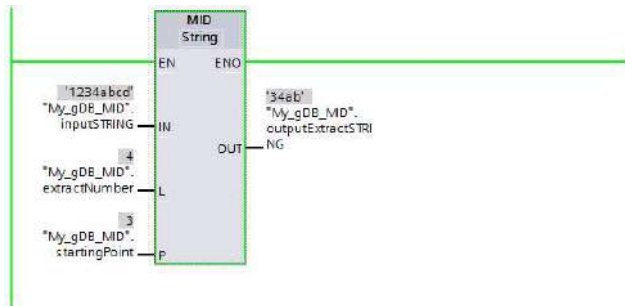
Create four tags in a global data block for storing the data.

My_gDB_MID			
	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	startingPoint	Int	3
4	extractNumber	Int	4
5	outputExtractSTRING	String	

Interconnect the parameters of the instruction as follows.



According to the value "4" of input parameter L ("extractNumber"), you extract a four-character partial character string from the character string ("inputSTRING"). The extraction starts from the third character ("startingPoint" has the value "3") of the character string ("inputSTRING"). The extracted partial character string is output at output parameter OUT ("outputExtractSTRING").



You can find additional information and the program code for the above named example here: [Sample Library for Instructions](#)

A.0.6 PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO LEFT

LEFT: Read the left character of a character string (S7-1200, S7-1500)

LEFT: Read the left character of a character string

Description

You use the instruction "LEFT" to extract a partial string beginning with the first character of the string at the IN input parameter. You specify the number of characters to be extracted in the L parameter. The extracted characters are output at the OUT output parameter in (W)STRING format.

If the number of characters to be extracted is greater than the current length of the character string, the OUT output parameter returns the input character string as a result. If the L parameter contains the value "0" or the input value is an empty string, an empty string will be returned. If the value in the L parameter is negative, an empty string will be output.

If errors occur during processing of the instruction and it is possible to write the OUT output parameter, an empty string will be output.

Parameters

The following table shows the parameters of the "LEFT" instruction:

Parameter	Declaration	Data type	Memory area	Description
IN	Input	STRING, WSTRING	D, L or constant	Character string
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L or constant	Number of characters to be extracted
OUT	Return	STRING, WSTRING	D, L	Extracted partial string

You can find additional information on valid data types under "[Overview of the valid data types](#)".

Example

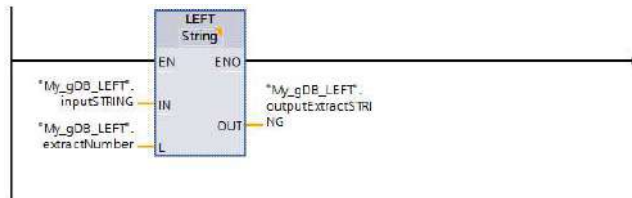
In the following example, you extract a partial character string with the STRING data type from a character string starting from the first character.

Create three tags in a global data block for storing the data.

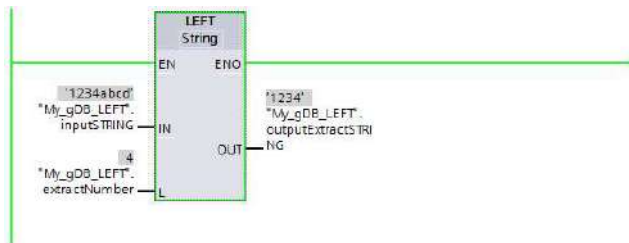
My_gDB_LEFT			
	Name	Data type	Start value
1	Static		
2	inputSTRING	String	'1234abcd'
3	extractNumber	Int	4
4	outputExtractSTRING	String	''

Interconnect the parameters of the instruction as follows.

LEFT: Read the left character of a character string (S7-1200, S7-1500)



According to the value "4" of parameter L ("extractNumber"), you extract a four-character partial character string from the character string ("inputSTRING") starting from the first character from the left. The extracted partial character string is output at output parameter OUT ("outputExtractSTRING").



You can find additional information and the program code for the above named example here: [Sample Library for Instructions](#)

A.0.7 PROGRAMAÇÃO SIEMENS S7 - FUNÇÃO CHARS_TO_STRG

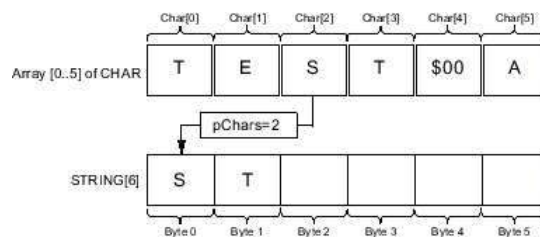
Chars_TO_Strg: Convert Array of CHAR to character string (S7-1200, S7-1500)

Chars_TO_Strg: Convert Array of CHAR to character string

Description

With the "Chars_TO_Strg" instruction you copy characters from an Array of CHAR or Array of BYTE to a character string of the data type STRING or from an ARRAY of WCHAR or Array of WORD to a character string of the data type WSTRING. For the copy operation only ASCII characters are valid.

- Specify the characters of the Array of (W)CHAR / BYTE / WORD that will be copied to a character string at the input parameter CHARS.
- The characters are written at the parameter STRG to a (W)STRING data type.
 - The number of characters in the string is at least the same number as were copied from the source field.
 - If the character string is shorter than the number of characters in the source field, the characters are written up to the maximum length of the string.
 - If the Array of CHAR / BYTE contains a "\$00" character or if the Array of WCHAR / WORD contains a W#16#0000 character, the copy operation is only carried out up to the corresponding position (see graphic).
- Using the PCHARS parameter, specify the position of the source field as of which the characters will be copied. PCHARS = 0 is the default value and always specifies the lower index limit of the array even if this is negative.
 - Example: if copying is to start with the third character of the source field, use the value "2" at parameter PCHARS:



- If an index is specified at parameter PCHARS and the index is not contained in the copy source (e.g. "7" at Array [0..5] of CHAR) then the instruction is not executed.

Note

Use of the instruction with S7-1200 V2.0

S7-1200 up to Version 2.0 only supports Array [0 .. n] of CHAR / BYTE. Negative index limits (e.g. Array [-3..2] of CHAR) are not permissible. This restriction is not checked by the software.

Parameters

The following table shows the parameters of the "Chars_TO_Strg" instruction:

Parameter	Declaration	Data type	Memory area	Description
CHARS	Input	VARIANT	D, L	Source of the copy operation

				Array of (W)CHAR / BYTE / WORD from which the characters are copied.
PCHARS	Input	DINT	I, Q, M, D, L, P or constant	Position in the Array of (W)CHAR / BYTE / WORD starting at which the characters are copied.
CNT	Input	UINT	I, Q, M, D, L, P or constant	Number of characters to be copied. Use "0" to copy all characters.
STRG	Output	STRING, WSTRING	D, L	<p>Destination of the copy operation</p> <p>Character string with the (W)STRING data type. Observe the maximum length of the data types:</p> <ul style="list-style-type: none"> STRING: 254 characters WSTRING: 254 characters (default) / 16382 characters (maximum) <p>When using WSTRING, note that you must define a length > 254 characters explicitly with square brackets (e.g., WSTRING[16382]).</p>

You can find additional information on valid data types under "[Overview of the valid data types](#)".

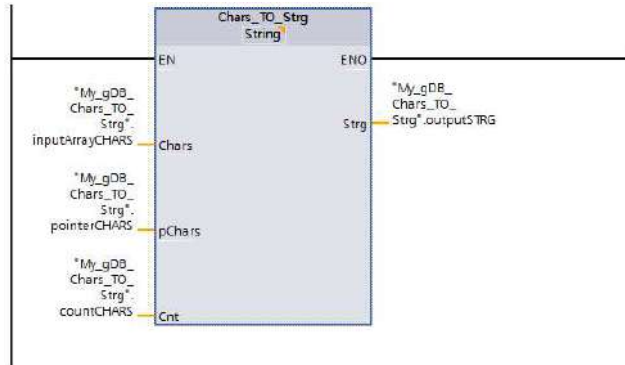
Example

In the following example, you copy characters of a structure of the Array of CHAR data type to a character string with the STRING data type.

Create four tags in a global data block for storing the data.

My_gDB_Chars_TO_Strg			
	Name	Data type	Start value
1	Static		
2	inputArrayCHARS	Array[0..9] of Char	
3	inputArrayCHARS[0]	Char	'M'
4	inputArrayCHARS[1]	Char	'Y'
5	inputArrayCHARS[2]	Char	'S'
6	inputArrayCHARS[3]	Char	'T'
7	inputArrayCHARS[4]	Char	'P'
8	inputArrayCHARS[5]	Char	'L'
9	inputArrayCHARS[6]	Char	'C'
10	inputArrayCHARS[7]	Char	''
11	inputArrayCHARS[8]	Char	''
12	inputArrayCHARS[9]	Char	''
13	pointerCHARS	DInt	2
14	outputSTRG	String	''
15	countCHARS	UInt	0

Interconnect the parameters of the instruction as follows and select the data type of the character string.



The CHARS structure ("inputArrayCHARS") is ten characters long (Array ... [0..9]). According to the value "2" of parameter PCHARS ("pointerCHARS"), characters are copied starting from the third position of the structure ("outputSTRG"). Starting from position "2", all characters of the structure ("inputArrayCHARS") are copied to the character string ("outputSTRG"), since parameter CNT ("countCHARS") has the value "0".

My_gDB_Chars_TO_Strg				
	Name	Data type	Start value	Monitor value
1	Static			
2	inputArrayCHARS	Array[0..9] of Char		
3	inputArrayCHARS[0]	Char	'M'	'M'
4	inputArrayCHARS[1]	Char	'y'	'y'
5	inputArrayCHARS[2]	Char	'S'	'S'
6	inputArrayCHARS[3]	Char	'7'	'7'
7	inputArrayCHARS[4]	Char	'P'	'P'
8	inputArrayCHARS[5]	Char	'L'	'L'
9	inputArrayCHARS[6]	Char	'C'	'C'
10	inputArrayCHARS[7]	Char	''	''
11	inputArrayCHARS[8]	Char	''	''
12	inputArrayCHARS[9]	Char	''	''
13	pointerCHARS	DInt	2	2
14	outputSTRG	String	''	'S7PLC '
15	countCHARS	UInt	0	0

You can find additional information and the program code for the above named example here: [Sample Library for Instructions](#)

A.0.8 CODIGO VBS - INSERT_SQL

Totally Integrated Automation Portal	
Project5 / PC-System_1 [SIMATIC PC station] / HMI_RT_1 [WinCC RT Advanced] / Scripts / VB scripts	
INSERT_SQL	
Comment	
0001	
0002	
0003	Sub INSERT_SQL()
0004	
0005	Dim conn, rst, SQL_Table
0006	Dim name_tag
0007	On Error Resume Next
0008	
0009	
0010	Set conn = CreateObject("ADODB.Connection")
0011	Set rst = CreateObject("ADODB.Recordset")
0012	
0013	'Open data source - Datenquelle öffnen
0014	conn.Open "Provider=MSDASQL;DSN=WIMCCSQL;UID=;PWD=;" 'DSN= Name of the ODBC database - DSN= Name der ODBC-Datenbank
0015	'Please replace all sequences which are enclosed with '_' by your own code.
0016	
0017	SQL_Table = "INSERT INTO WINCC_TEST.dbo.TEST VALUES ('"& SmartTags("Data_block_1_DATA_STRING") &"')"
0018	Set rst = conn.Execute(SQL_Table)
0019	conn.commit()
0020	
0021	'Close data source - Datenquelle schließen
0022	conn.close
0023	
0024	Set rst = Nothing
0025	Set conn = Nothing
0026	
0027	
0028	End Sub

A.0.9 CODIGO VBS - INSERT_OBJECT

Totally Integrated Automation Portal		
Project5 / PC-System_1 [SIMATIC PC station] / HMI_RT_1 [WinCC RT Advanced] / Scripts / VB scripts		
INSERT_OBJECT		
Comment		
0001		
0002		
0003	Sub INSERT_OBJECT()	
0004		
0005		
0006	Dim conn, rst, SQL_Table	
0007	Dim name_tag	
0008	On Error Resume Next	
0009		
0010		
0011	Set conn = CreateObject("ADODB.Connection")	
0012	Set rst = CreateObject("ADODB.Recordset")	
0013		
0014	'Open data source - Datenquelle öffnen	
0015	conn.Open "Provider=MSDASQL;DSN=WIMCCSQL;UID=;PWD=;" 'DSN= Name of the ODBC database - DSN= Name der ODBC-Datenbank	
0016	'Please replace all sequences which are enclosed with '_' by your own code.	
0017		
0018	SQL_Table = "INSERT INTO WINCC_TEST.dbo.TABLE_OBJECTS VALUES ('& SmartTags("ID_OBJECTS") &"', '& SmartTags("Data_block_1_DATA_INSERT_STRING") &"')"	
0019	Set rst = conn.Execute(SQL_Table)	
0020	conn.commit()	
0021		
0022	'Close data source - Datenquelle schließen	
0023	conn.close	
0024		
0025	Set rst = Nothing	
0026	Set conn = Nothing	
0027		
0028		
0029		
0030	End Sub	

A.0.10 MANUAL DE SINTAXE SQL

18/01/2021

SQL Quick Reference

SQL Statement	Syntax
AND / OR	SELECT column_name(s) FROM table_name WHERE condition AND OR condition
ALTER TABLE	ALTER TABLE table_name ADD column_name datatype or ALTER TABLE table_name DROP COLUMN column_name
AS (alias)	SELECT column_name AS column_alias FROM table_name or SELECT column_name FROM table_name AS table_alias
BETWEEN	SELECT column_name(s) FROM table_name WHERE column_name BETWEEN value1 AND value2
CREATE DATABASE	CREATE DATABASE database_name
CREATE TABLE	CREATE TABLE table_name (column_name1 data_type, column_name2 data_type, column_name3 data_type, ...)
CREATE INDEX	CREATE INDEX index_name ON table_name (column_name) or CREATE UNIQUE INDEX index_name ON table_name (column_name)
CREATE VIEW	CREATE VIEW view_name AS SELECT column_name(s) FROM table_name WHERE condition
DELETE	DELETE FROM table_name WHERE some_column=some_value

https://www.w3schools.com/sql/sql_quickref.asp

1/4

	<p>or</p> <p>DELETE FROM table_name (Note: Deletes the entire table!!)</p> <p>DELETE * FROM table_name (Note: Deletes the entire table!!)</p>
DROP DATABASE	DROP DATABASE database_name
DROP INDEX	<p>DROP INDEX table_name.index_name (SQL Server)</p> <p>DROP INDEX index_name ON table_name (MS Access)</p> <p>DROP INDEX index_name (DB2/Oracle)</p> <p>ALTER TABLE table_name DROP INDEX index_name (MySQL)</p>
DROP TABLE	DROP TABLE table_name
EXISTS	<p>IF EXISTS (SELECT * FROM table_name WHERE id = ?)</p> <p>BEGIN</p> <p>--do what needs to be done if exists</p> <p>END</p> <p>ELSE</p> <p>BEGIN</p> <p>--do what needs to be done if not</p> <p>END</p>
GROUP BY	<p>SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name</p>
HAVING	<p>SELECT column_name, aggregate_function(column_name) FROM table_name WHERE column_name operator value GROUP BY column_name HAVING aggregate_function(column_name) operator value</p>
IN	<p>SELECT column_name(s) FROM table_name WHERE column_name IN (value1,value2,..)</p>
INSERT INTO	<p>INSERT INTO table_name VALUES (value1, value2, value3,....)</p> <p>or</p> <p>INSERT INTO table_name (column1, column2, column3,...) VALUES (value1, value2, value3,....)</p>
INNER JOIN	SELECT column_name(s)

	<pre>FROM table_name1 INNER JOIN table_name2 ON table_name1.column_name=table_name2.column_name</pre>
LEFT JOIN	<pre>SELECT column_name(s) FROM table_name1 LEFT JOIN table_name2 ON table_name1.column_name=table_name2.column_name</pre>
RIGHT JOIN	<pre>SELECT column_name(s) FROM table_name1 RIGHT JOIN table_name2 ON table_name1.column_name=table_name2.column_name</pre>
FULL JOIN	<pre>SELECT column_name(s) FROM table_name1 FULL JOIN table_name2 ON table_name1.column_name=table_name2.column_name</pre>
LIKE	<pre>SELECT column_name(s) FROM table_name WHERE column_name LIKE pattern</pre>
ORDER BY	<pre>SELECT column_name(s) FROM table_name ORDER BY column_name [ASC DESC]</pre>
SELECT	<pre>SELECT column_name(s) FROM table_name</pre>
SELECT *	<pre>SELECT * FROM table_name</pre>
SELECT DISTINCT	<pre>SELECT DISTINCT column_name(s) FROM table_name</pre>
SELECT INTO	<pre>SELECT * INTO new_table_name [IN externaldatabase] FROM old_table_name or SELECT column_name(s) INTO new_table_name [IN externaldatabase] FROM old_table_name</pre>
SELECT TOP	<pre>SELECT TOP number percent column_name(s) FROM table_name</pre>
TRUNCATE TABLE	<pre>TRUNCATE TABLE table_name</pre>
UNION	<pre>SELECT column_name(s) FROM table_name1 UNION</pre>

	SELECT column_name(s) FROM table_name2
UNION ALL	SELECT column_name(s) FROM table_name1 UNION ALL SELECT column_name(s) FROM table_name2
UPDATE	UPDATE table_name SET column1=value, column2=value,... WHERE some_column=some_value
WHERE	SELECT column_name(s) FROM table_name WHERE column_name operator value