



Braço robótico colaborativo de baixo custo

Versão final após defesa

Estevão Dos Prazeres Ananias

Dissertação para obtenção do Grau de Mestre em
Engenharia Eletrotécnica de Computadores
(2º ciclo de estudos)

Orientador: Prof. Doutor Pedro Miguel de Figueiredo Dinis Oliveira Gaspar

julho de 2022

Agradecimentos

Muitas pessoas são responsáveis pela qualidade deste trabalho. Agradecer a todos sem exceção que contribuíram direta ou indiretamente para que este projeto tornasse uma realidade.

Resumo

A evolução das tecnologias de informação e os grandes avanços na área da inteligência artificial, estão a conduzir a um nível de automatização até agora nunca alcançado. Grande parte desta automatização deve-se ao recurso a robôs. Apesar dos primeiros robôs industriais terem surgido na década de 60 e serem amplamente aceites como uma forma de melhoria da qualidade e da produtividade, as questões como segurança dos humanos, equipamentos e elevado custo para aquisição desta tecnologia, faziam com que esses robôs devessem operar dentro de células de fabrico para garantir a segurança ao seu redor. Dada as necessidades dos robôs colaborarem em simultâneo ou em paralelo com os humanos, diante desses desafios, uma nova geração de robô, designados Cobots, “*Collaborative Robots*”, está a ganhar destaque. Com os Cobots é possível ultrapassar as barreiras de segurança e perspetivar um trabalho lado a lado com os humanos em segurança. Ainda assim, e tal como até então, o custo destas tecnologias é demasiado elevado para uma grande maioria das empresas.

Este trabalho apresenta o desenvolvimento e teste de um braço robótico colaborativo articulado com 6 eixos, de baixo custo, dentro dos padrões standardizados, podendo ser utilizado em diferentes aplicações. O desenvolvimento deste robô colaborativo requereu o projeto e impressão 3D da estrutura (elos e peças), dimensionamento e seleção dos circuitos e/ou componentes eletrónicos, programação e controlo. No âmbito do trabalho, foi também desenvolvida de uma aplicação computacional e respetivo interface com utilizador para definição de sequências de movimentos e/ou para controlo remoto. A solução tecnológica desenvolvida nesta dissertação permite estender ainda mais a aplicação da robótica colaborativa a setores e aplicações, industriais, de serviços e até mesmo educacionais, que por questões dos custos associados se encontram restritas. Em particular, no âmbito de aplicações educacionais, o robô colaborativo e a ferramenta computacional de programação, permitem consolidar o conhecimento nos conceitos fundamentais associados à cinemática e dinâmica do movimento de braços robóticos, assim como na mecânica, eletrónica e controlo inerente aos componentes que compõem a solução tecnológica.

Palavras-chave

Robô, Robô colaborativo, Segurança, Humanos, Tecnologias, Circuito eletrónico, Controlo, Programação, PLA, impressão 3D.

Abstract

The evolution of information technology and the great advances in artificial intelligence are leading to a level of automation that has never been reached before. A large part of this automation is due to the use of robots.

Although the first industrial robots appeared in the 1960s and were widely accepted as a way to improve quality and productivity, issues such as human safety, equipment and the high cost of acquiring this technology meant that these robots had to operate inside manufacturing cells to ensure safety around them.

Given the needs of robots to collaborate simultaneously or in parallel with humans, in the face of these challenges, a new generation of robots, called Cobots, "Collaborative Robots", are gaining prominence. With Cobots it is possible to overcome security barriers and to envisage working safely side by side with humans. Still, as before, the cost of these technologies is too high for most companies.

This work aims to develop and test a low-cost 6-axis articulated collaborative robotic arm, within standardized standards, which can be used in different applications. The development of this collaborative robot involves the design and 3D printing of the structure (links and parts), sizing and selection of circuits and/or electronic components, programming, and control. Also, part of this work is the development of a computational application and respective user interface for defining movement sequences and/or for remote control. The technological solution developed in this dissertation makes it possible to further extend the application of collaborative robotics to industrial, service and even educational sectors and applications, which due to the associated costs are restricted. In the scope of educational applications, the collaborative robot and the computational programming tool allow the consolidation of knowledge in the fundamental concepts associated with the kinematics and dynamics of the movement of robotic arms, as well as in the mechanics, electronics, and control inherent to the components that compose the technological solution.

Keywords

Robot, Collaborative Robot, Safety, Human, Technologies, Electronic circuit, Control, Programming, PLA, 3D printing.

Índice

Agradecimentos	i
Resumo	iii
Abstract	v
Índice	vii
Lista de Figuras	ix
Lista de Tabelas	xi
Nomenclatura	xiii
1. Introdução	1
1.1. Enquadramento	1
1.2. O problema em estudo e a sua relevância	3
1.3. Objetivos e contribuição da dissertação	4
1.4. Visão geral e organização da dissertação.....	4
2. Estado da Arte	7
2.1. Introdução.....	7
2.2. A evolução da robótica.....	8
2.2.1. Primeiro robô industrial	8
2.2.2. Primeiro robô colaborativo	9
2.3. Robô colaborativo comercializáveis.....	9
2.4. Robôs baixos custos	14
2.5. Nota conclusiva.....	23
3. Materiais e Métodos	25
3.1. Materiais.....	25
3.1.1. Hardware	25
3.1.1.1. Impressão 3D	25
3.1.1.2. Controlo e movimento.....	28
3.1.1.3. Microcontrolador	29
3.1.2. Software	34
3.1.2.1. Software de desenho assistido por computador - SolidWorks.....	34
3.1.2.2. Ambiente de Desenvolvimento Integrado Arduíno IDE.....	34
3.1.2.3. MATLAB - Matrix Laboratory	35
3.2. Métodos.....	35
3.2.1. Matriz de transformação Homogénea.....	35

3.2.2.	Cinemática direta	37
3.2.3.	Ângulo de Euler	38
3.2.4.	Cinemática inversa	39
3.2.5.	Velocidade e Jacobiana	40
3.2.6.	Singularidade	43
3.2.7.	Rede Neuronal Artificial	43
3.3.	Nota conclusiva.....	44
4.	Protótipo e Dimensionamento do Braço Robótico	47
4.1.	Peças e estrutura do braço robótico	47
4.2.	Formulação matemática da convenção – DH e cinemática direta/inversa	49
4.3.	Circuito elétrico, controlo e treino da rede neuronal	56
4.4.	Interface gráfico do utilizador - MATLAB.....	68
4.5.	Impressões 3D e montagem do braço robótico	72
4.6.	Nota conclusiva.....	76
5.	Análise de Discussão de Resultados	77
5.1.	Teste N.º 1: Carga útil.....	77
5.2.	Teste N.º 2: Erro de posicionamento	79
5.3.	Teste N.º 3: Precisão e repetibilidade	83
5.4.	Teste N.º 4: Verificação da cinemática inversa	93
5.5.	Teste N.º 5: Corrente elétrica vs. Carga no efector.....	93
5.6.	Teste N.º 6: Detecção de colisões	95
5.7.	Nota conclusiva.....	96
6.	Conclusões	97
6.1.	Conclusões gerais	97
6.2.	Sugestões de trabalhos futuros	98
	Referências Bibliográficas	99
	Manual do utilizador	105

Lista de Figuras

Fig. 1. Números (x1000) dos robôs industriais instalado anualmente (IFR, 2020).	1
Fig. 2. Evolução anual da robótica industrial em diferentes sectores (IFR, 2020).	2
Fig. 3. Evolução anual da aquisição (x1000) de robôs colaborativos face aos robôs tradicionais (IFR, 2020).	2
Fig. 4. Algumas configurações dos manipuladores robóticos (IFR, 2016).	7
Fig. 5. Primeiro robô industrial, Unimate (ReachRobotics, 2018).	8
Fig. 6. Robô LBR iiwa 7 R800 e seu envelope de trabalho (KUKA, 2014).	10
Fig. 7. Robô UR5e e interface gráfico para efetuar movimentos (Universal Robots, 2012).	11
Fig. 8. Robô ABB YuMi IRB 14000 e seu envelope de trabalho (ABB, 2021).	12
Fig. 9. Robô FANUC CR-4iA e seu envelope de trabalho (FANUC, 2020).	13
Fig. 10. Robô HydraX, dimensões e vista frontal do envelope do trabalho (Krimpenis <i>et al.</i> , 2020).	14
Fig. 11. Robô antropomórfico com elo rígido variável e funcionamento de VSL (Stilli <i>et al.</i> , 2017).	15
Fig. 12. Robô WE-R2.4 e a garra (Thingiverse, 2020).	16
Fig. 13. Duplo robô SCARA e área do envelope do trabalho (Liu <i>et al.</i> , 2018).	17
Fig. 14. Robô humanóide leve (Lightweight Humanoid - LWH) (Yang <i>et al.</i> , 2019).	18
Fig. 15. Braço robótico com materiais recicláveis e diagrama de circuito de controlo (Karmoker <i>et al.</i> , 2014).	19
Fig. 16. Estrutura de braço robótico com deteção baseada em visão computacional e seu circuito de controlo (Sundaram <i>et al.</i> , 2019).	20
Fig. 17. Estrutura, projeto e montagem do braço robótico macio (Arm <i>et al.</i> , 2018).	21
Fig. 18. Resultado do angulo de curvatura e a força suportada sob pressão do braço robótico macio (Arm <i>et al.</i> , 2018).	22
Fig. 19. Fluxograma e protótipo real do sistema de controlo e atuação do braço robótico macio (Arm <i>et al.</i> , 2018).	22
Fig. 20. Processos para impressão 3D e visão dos tipos de materiais e métodos de fabricação aditivas (Dizon <i>et al.</i> , 2018).	26
Fig. 21. Representação da impressora 3D.	27
Fig. 22. Diagrama do disco codificador dos encoder.	29
Fig. 23. Diagrama de bloco de função do ESP2 (Espressif, 2021).	30
Fig. 24. Estrutura e ligação para comunicação UART.	31

Fig. 25. Diferentes configurações de comunicação com o protocolo ESP NOW (Gnd_To_Vcc, 2021).	33
Fig. 26. Representação estrutural da comunicação I2C (Bagdalkar & Ali, 2020).	33
Fig. 27. Representação da matriz homogénea.	35
Fig. 28. Representação do manipulador plano com duas juntas rotacionais (Spong <i>et al.</i> , 2006).	43
Fig. 29. Arquitetura da rede neuronal artificial (Fernández <i>et al.</i> , 2014).	44
Fig. 30. Peças e estrutura 3D do braço robótico.	48
Fig. 31. Cinemática do manipulador robótico.	49
Fig. 32. Pseudocódigo para meta heurística PSO.	54
Fig. 33. Esquema e tipologia de comunicação entre os ESP32 e PC.	57
Fig. 34. Esquema de comunicação entre o ESP32 principal e PC.	57
Fig. 35. Esquema de ligação entre os dispositivos de cada eixo.	58
Fig. 36. Placa circuito impresso (PCB) desenvolvida com componentes soldados.	59
Fig. 37. Rampa de números de pulsos aplicar ao motor em função de erro.	60
Fig. 38. Estrutura de rede neuronal artificial criada.	66
Fig. 39. Evolução do treinamento da rede neuronal com o número de iteração.	67
Fig. 40. Comparação entre corrente medida e estimada pela rede neuronal.	68
Fig. 41. Interface do utilizador, menu “Conectar”	69
Fig. 42. Interface do utilizador, menu “Mover”	69
Fig. 43. Interface do utilizador, menu “Alarme”.	70
Fig. 44. Interface do utilizador, menu “Controlo”	71
Fig. 45. Interface do utilizador, menu “Programa”.	71
Fig. 46. Impressões e processo evolutivo da montagem do braço robótico.	72
Fig. 47. Robô posicionado para teste n.º 1 de carga útil.	78
Fig. 48. Robô em diferentes posições durante o teste n.º 2 no posicionamento.	80
Fig. 49. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade. Erro!	
Marcador não definido.	
Fig. 49. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).	84
Fig. 49. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).	85
Fig. 49. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).	86
Fig. 50. Área de trabalho com posições e sequência de pontos para o teste de precisão e de repetibilidade.	87
Fig. 51. Resultado da corrente em função de peso obtido durante teste n.º 5.	94
Fig. 52. Resultado da corrente máxima estimada e medida durante o teste n.º 6.	95

Lista de Tabelas

Tabela 1. Especificação técnica do robô LBR iiwa 7 R800 (KUKA, 2014).....	10
Tabela 2. Especificações técnicas do Robô UR5e (Universal Robots, 2012).....	11
Tabela 2. Especificações técnicas do Robô UR5e (Universal Robots, 2012) (cont.).....	12
Tabela 3. Especificações técnicas do robô YuMi IRB 14000 (ABB, 2021).....	13
Tabela 4. Especificações técnicas do robô FANUC CR-4iA (FANUC, 2020).....	14
Tabela 5. Especificações técnicas do robô HydraX.....	15
Tabela 6. Especificações técnicas do robô SCARA com duplo braço (Liu <i>et al.</i> , 2018).....	17
Tabela 7. Amplitude dos movimentos do robô humanoide leve (Yang <i>et al.</i> , 2019).....	18
Tabela 8. Ensaios efetuado com o braço robótico com detecção baseada em visão computacional (Sundaram <i>et al.</i> , 2019).....	20
Tabela 9. Propriedades dos filamentos mais utilizados na impressão 3D.....	27
Tabela 10. Valores das distâncias entre elos.....	49
Tabela 11. Parâmetros DH do manipulador robótico.....	49
Tabela 12. Valores das resistências utilizada.....	63
Tabela 13. Parâmetro velocidade (Pulso Por Segundo [PPS]) aplicar aos motores.....	65
Tabela 14. Resultado de treino da rede neuronal artificial.....	67
Tabela 15. Parâmetros principais de impressão para diferente filamento.....	72
Tabela 16. Ângulos, posições e orientações escolhido para teste de posicionamento.....	82
Tabela 17. Ângulos, posições e orientações, resultado do teste 1 de posicionamento.....	82
Tabela 18. Erro médio quadrático associado ao teste 1.....	83
Tabela 19. Ângulos, posições e orientações escolhido para teste de repetibilidade.....	87
Tabela 20. Ângulos de cada junta registada durante o teste de repetibilidade e precisão.....	89
Tabela 21. Posição do efector registada durante o teste de repetibilidade e precisão.....	90
Tabela 22. Orientação do efector registada durante o teste de repetibilidade e precisão.....	91
Tabela 23. Precisão e repetibilidade obtida durante os testes.....	92
Tabela 24. Resultado da eficiência no cálculo da cinemática inversa.....	93

Nomenclatura

Geral:

R	Resistência elétrica, [Ω];
t	Tempo, [s];
T	Temperatura, [K ou $^{\circ}\text{C}$];
V	Diferença de potencial, [V];
I	Corrente, [A];
\dot{Q}	Potência térmica, [W];
J	Jacobiana;

Simbologia grega:

θ	Ângulo da junta, [$^{\circ}$];
----------	----------------------------------

Acrónimos:

ISO	International Organization for Standardization.
COBOT	Collaborative robot.
IFR	International federation of robotics.
DOF	Degree of freedom.
TCP	Tool Center Point.
PET	Polyethylene terephthalate.
PLA	Polylactic Acid.
AM	Additive Manufacturing.
MENA	National Electronic Manufacturers Association.
3D	Tridimensional.
FDM	Fused Deposition Modeling.
STL	Standard Triangle Language.
PCB	Printed Circuit Board.
RP	Pose Repeatability.
H	Homogeneous Matrix.

ESP	ESPRESSIF.
UR	Universal Robots.
ABB	Asea Brown Boveri.
KUKA	Keller Und Knappich Augsburg.
FANUC	Fuji Automatic Numerical Control.
SCARA	Selective Compliance Assembly Robot Arm.
PPS	Pulse Per Second.
UR	Universal Robots.
UART	Universal Asynchronous Receiver Transmitter.
I2C	Inter Integrated Circuit.
MAC	Media Access Control.
IoT	Internet off Things.
PSO	Particle Swarm Optimization.
NNA	Neural Network Algorithm.
DH	Denavit Hartenberg.
IDE	Integrated Development Environment.
GUIDE	Graphical User Interface Development Environment.
CAD	Computer Aided Design.
ADC	Analogic Digital Converter.
MATLAB	Matrix Laboratory.
RMSE	Root Mean Square Error.
CMRR	Common Mode Rejection Ratio.
RMSE	Root Mean Square Error.
USB	Universal Serial Bus.
PC	Personal Computer.

1. Introdução

1.1. Enquadramento

Segundo a literatura está-se perante a quarta revolução industrial, caracterizada por avanços em tecnologias emergentes em áreas como a robótica, inteligência artificial, nanotecnologia, computação quântica, internet das coisas, geração de tecnologia sem fio e veículos autónomos. Nesta fase as máquinas estão conectadas e comunicam entre si, permitindo que os sistemas tomam decisões sem ação humana.

A Indústria 4.0 é a grande responsável pelas transformações em toda a cadeia produtiva, desde as áreas de projetos até a estrutura interna das empresas. Tecnologia e robótica movimentam os mercados, fornecendo soluções capazes de elevar a produtividade e estimular os negócios. A maior vantagem dos robôs é que podem ser usados em atividades perigosas e insalubres, permitindo que os profissionais coordenem as tarefas a partir de um ambiente seguro, sem que haja risco para a saúde.

De acordo com a *International Federation of Robotics* (IFR, 2020) as vendas de robôs industriais permanecem em alto nível, tendo sido instaladas 373.000 unidades a nível global, conforme indicado na Fig. 1. Prevê-se um aumento médio de 12% entre 2020 e 2022. Na Fig. 2 são apresentados alguns sectores nos quais a robótica industrial é aplicada.

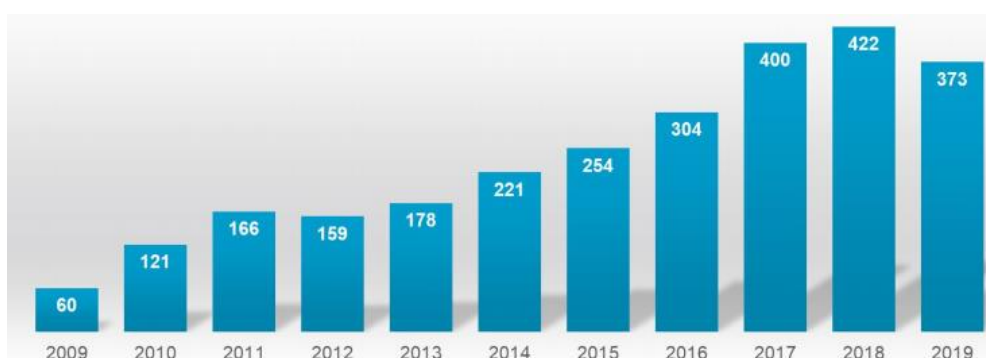


Fig. 1. Números (x1000) dos robôs industriais instalado anualmente (IFR, 2020).

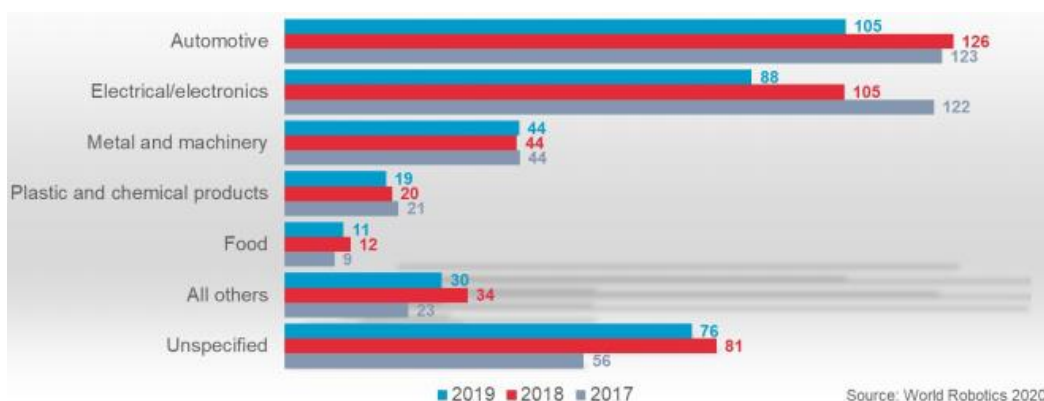


Fig. 2. Evolução anual da robótica industrial em diferentes sectores (IFR, 2020).

A robótica não se aplica apenas às indústrias, também é aplicada a serviços. De acordo com (IFR, 2020), a robótica de serviços atingiu níveis de venda em 2019 muito relevantes : (1) 23,2 milhões de unidades de robôs de serviço pessoal/doméstico (tarefas domésticas e de entretenimento), representando um aumento de 34% em relação ao ano anterior; (2) 173.000 unidades de robôs de serviços profissionais (logística, ambientes públicos, aplicação defesa, inspeção e manutenção, limpeza profissional, agrícolas, exoesqueletos, aplicação médica e construção ou demolição), representando um aumento de 32% em relação ao ano anterior.

A robótica é caracterizada como uma área muito multidisciplinar e vasta, na qual, para além das partes acima apresentadas, surgiu um novo tema a qual é chamado de Robótica Colaborativa. Este tema tem ganha destaque pela sua grande aplicabilidade e simplicidade. Na Fig. 3 pode-se observar a evolução de robôs colaborativos em relação aos robôs tradicionais.

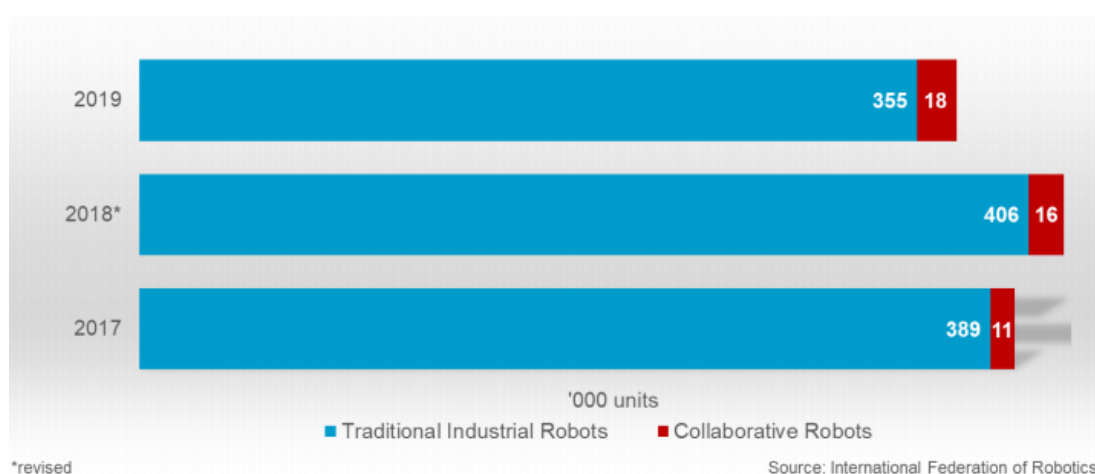


Fig. 3. Evolução anual da aquisição (x1000) de robôs colaborativos face aos robôs tradicionais (IFR, 2020).

1.2. O problema em estudo e a sua relevância

Com os avanços tecnológicos, aumento da população a nível mundial (necessidade de produção e serviços em grande escala), maior tendência em melhorar e otimizar a qualidade de produtos e/ou operações (devido à incerteza, erros e à sensibilidade humana), devido às tarefas não favoráveis e também à necessidade de redução de acidentes em algumas aplicações, muitos sectores industriais estão a optar cada vez mais por aplicações robóticas para resolver e executar tarefas. Deste modo, os robôs tornam-se num mecanismo muito importante para indústrias pela vasta aplicação, rápida adaptação, e trabalho contínuo.

Quando se fala do termo robótica, normalmente surgem questões como segurança, flexibilidade e custo. A *Robotic Industries Association* (RIA, 2019) apresenta os benefícios dos robôs colaborativos face a estas questões. Quanto à segurança, os robôs colaborativos são projetados para minimizar o risco de acidentes e lesões no local de trabalho, estando equipados com sensores para evitar colisões, limitações de força, proteção de sobrecarga (aumento da corrente elétrica) e conformidade passiva em caso de contato não planeado. Estas normas de segurança segue as normas padrão (ISO 15066, 2016) . Quanto à flexibilidade, os robôs colaborativos podem ser facilmente programados, mesmo por trabalhadores sem conhecimento de programação de robôs. Quanto ao custo, a facilidade de programação de um robô colaborativo reduz o tempo e os recursos necessários para sua integração no processo, o que diminui o investimento em automação. Os robôs colaborativos vêm equipados com recursos de segurança e não exigem cercas ou outros equipamentos de segurança industrial, o que reduz ainda mais os custos e o tempo de integração. Os fatores que influenciam na aquisição de um robô colaborativo são internos (estrutura, tamanho, infraestrutura, etc.), externos (fornecedores, concorrentes, agências de financiamentos e governamentais, clientes, etc) e tecnológicos (análise económica, grau de inovação, compatibilidade, velocidade para realizar tarefas, complexidade, confiabilidade) (Simões *et al.*, 2020). A interação humano-robô poderá combinar o espaço de trabalho em diferentes formas e conseqüentemente diferentes classificações (Zacharaki *et al.*, 2020), nomeadamente:

- Coexistência: o humano e o robô realizam a sua atividade lado a lado, mas não partilham o mesmo espaço de trabalho;
- Sincronização: o humano e o robô partilham o mesmo espaço de trabalho, mas apenas uma destas partes está realmente presente no espaço de trabalho a qualquer momento;
- Cooperação: o humano e o robô partilham o mesmo espaço de trabalho, sendo que as duas partes podem executar tarefas ao mesmo período, contudo não funcionam simultaneamente no mesmo produto ou componente;

- Colaboração: o humano e o robô partilham o mesmo espaço de trabalho, sendo que as duas partes podem executar tarefas ao mesmo tempo no mesmo e funcionam simultaneamente no mesmo produto ou componente;

1.3. Objetivos e contribuição da dissertação

O objetivo desta dissertação consiste no desenvolvimento de um braço robótico colaborativo de baixo custo. O desenvolvimento de uma solução desta natureza, permite aplicar o robô a atividades laboratoriais de ensino-aprendizagem no âmbito de unidades curriculares de cursos de Engenharia, apoio a atividades de investigação no âmbito de projetos, e até mesmo testar a sua adequação a outro tipo de aplicação, fora deste contexto, na indústria ou serviços.

Em particular a utilização do braço robótico desenvolvido no âmbito das unidades curriculares de robótica é uma mais-valia para os alunos terem oportunidade de desenvolver trabalhos práticos e enriquecer os conhecimentos prático. Por outro lado, a aplicação em projeto de investigação pode ser variada, mas especificamente pode ser associada a projetos de robotização de tarefas agrícolas, nomeadamente na apanha de frutos caídos no solo. A recolha de frutos caídos de forma automática terá vários impactos, tanto diretos como indiretos, nas culturas, e em particular naquela que envolve um produto endógeno da região, a cultura do pêssego. Em resultado da queda das árvores, os frutos ficam depositados no solo potenciando diversos problemas, nomeadamente o desenvolvimento de doenças ou o completar de ciclo de algumas pragas.

1.4. Visão geral e organização da dissertação

Esta dissertação está, essencialmente organizado em seis capítulos: o primeiro capítulo consiste na introdução à robótica em geral e posteriormente à robótica colaborativa, versando a evolução da robótica em geral a nível mundial, sendo posteriormente descrito o problema em estudo e a sua relevância, assim como o objetivo e contributo da dissertação.

O segundo capítulo descrevo o estado arte, onde é providenciada uma abordagem da robótica em geral e apresentadas algumas soluções tecnologias e fabricantes existente no mercado.

O terceiro capítulo apresenta os materiais e métodos para desenvolvimento do braço robótico colaborativo de baixo custo.

O quarto capítulo inclui o desenvolvimento, controlo e testes ao braço projetado.

O quinto capítulo fornece uma análise e discussão de resultados ao protótipo desenvolvido.

O último capítulo, o sexto capítulo, apresenta as conclusões ao trabalho desenvolvido e algumas sugestões de trabalho futuro.

2. Estado da Arte

Neste capítulo é apresentada uma visão sobre a robótica em geral, relativamente à sua constituição, tipologias, e demais configurações existentes. Posteriormente são apresentados alguns robôs colaborativo existente no mercado e os seus respectivos fabricante e aplicações, soluções de baixo custos, presente na literatura como caso de estudo e/ou investigação e também uma solução ao qual se tenciona replicar a sua estrutura física.

2.1. Introdução

Um robô colaborativo é um dispositivo robótico que manipula objetos e trabalha diretamente ao lado de humanos, sem cercas de segurança ou barreiras (Simões *et al.*, 2020). O que diferencia um robô, é fundamentalmente a sua estrutura mecânica. Quer a sua base seja móvel ou não, os robôs podem ser categorizados em robôs manipuladores ou robôs móveis. A estrutura consiste numa sequência de corpos rígidos, os chamados elos, interligados por meio de articulações prismáticas (P) ou rotacionais (R), as chamadas juntas, permitindo translação e/ou rotação segundo um referencial de coordenadas X , Y , e Z . A estrutura mecânica, chamada de braço, visa posicionar um punho para que a ferramenta fique orientado na área de trabalho. Dependendo da disposição dos graus de liberdade (DOF - *Degrees Of Freedom*) do braço, diferentes configurações de manipuladores robóticos podem ser projetadas. As principais configurações são (a) articulada, (b) cartesiana, (c) SCARA (*Selective Compliance Assembly Robot Arm*) e (d) delta (IFR, 2016), representado na Fig. 4. Estas configurações combinam entre elas, permitindo com que o robô executa movimento linear, circular e juntas.

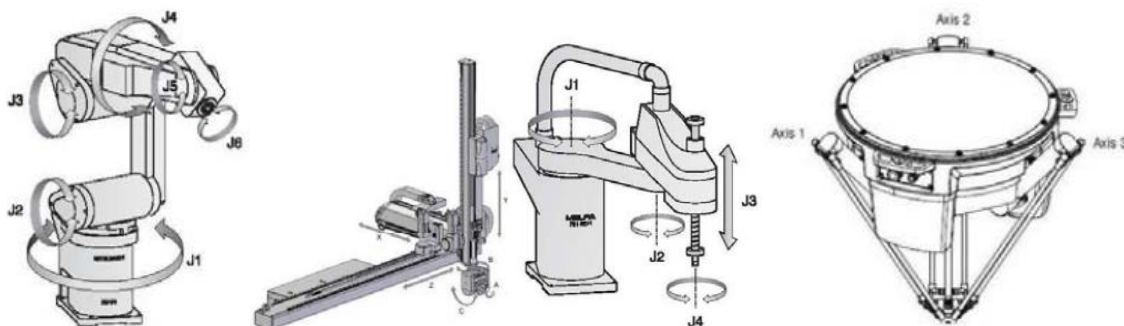


Fig. 4. Algumas configurações dos manipuladores robóticos (IFR, 2016).

2.2. A evolução da robótica

2.2.1. Primeiro robô industrial

O termo robô utilizado nos dias de hoje, advém da palavra “*Robota*”, que significa “trabalho pesado” ou “trabalho forçado”. Este termo surgiu deve-se ao escritor checo Karel Capek (1890-1938), em 1920 no seu romance “*RUR, Rossum's Universal Robots*”. Por outro lado, a palavra “Robótica” foi empregue pela primeira vez por **Isaac Asimov** (1920-1992) no seu romance “*Runaround*” (1942), contido na série “*I, Robot*”. Na década de 1950, **George Charles Devol Jr.** (1912-2011), surgiu com o conceito de robôs industriais, automatizando tarefas simples e repetitivas. Em 1954, associou-se a Joseph Engelberger (1925 - 2015) e fundaram a empresa “Unimation”, que produziu em 1961 o primeiro robô “Unimate” apresentado na Fig. 5. Este robô foi instalado na indústria automóvel, nomeadamente na fábrica da General Motors situada em Trenton (New Jersey, EUA) e foi utilizado para extrair peças de uma máquina de fundição sob pressão (A. Gasparetto & Scalera, 2019).

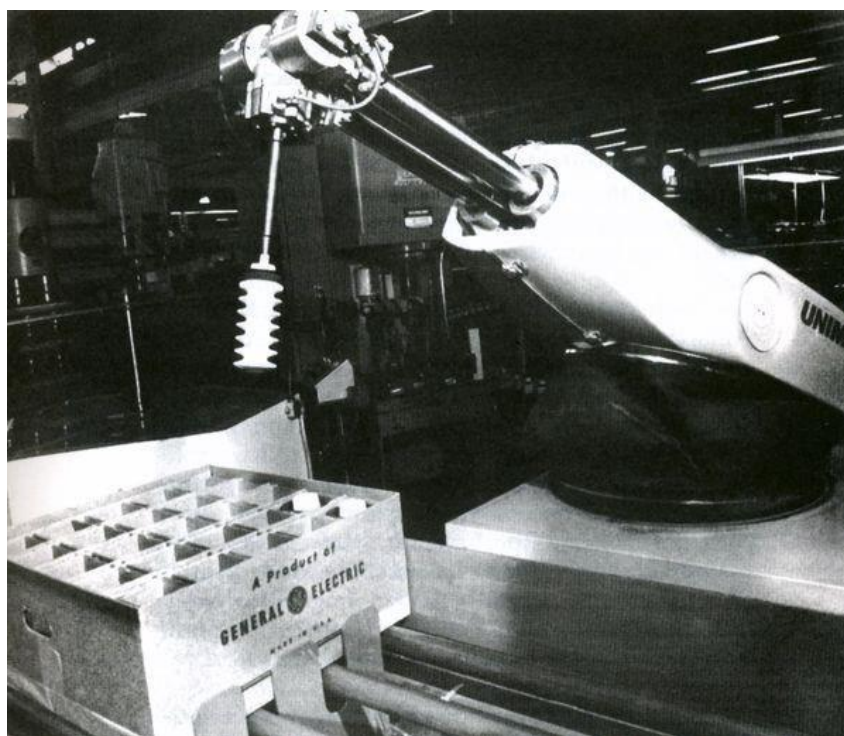


Fig. 5. Primeiro robô industrial, Unimate (ReachRobotics, 2018).

O robô Unimate possuía as seguintes características: 6 graus de liberdade, atuadores hidráulicos, peso de 1575 kg, repetibilidade 1 mm, sensores codificadores rotativos personalizados, base de aço e braço de alumínio (*RoboT Unimate*, 2018).

2.2.2. Primeiro robô colaborativo

O termo Cobot surgiu em 1996 através de James E. Colgate e Michael A. Peshkin e posteriormente foi submetida uma patente americana em 1999 intitulada sobre “Cobot”, que descreve "um aparelho e método para interação entre humano e manipulador robótico de uso geral controlado por um computador" (Colgate & Peshkin, 1999).

Em 2004, a KUKA acrescentou outra inovação mundial com o desenvolvimento de primeiro Cobot do mundo, robô LBR 3 (LightWeight Robot ou LBR). O KUKA LBR 3 era diferente dos outros robô industriais porque permitia a interação humana direta sem barreiras de segurança.

Em 2008, o fabricante dinamarquês Universal Robots lançou o UR5, um Cobot que pode operar com segurança ao lado dos humanos, eliminando a necessidade de grades ou cercas de segurança (Hand, 2020).

2.3. Robô colaborativo comercializáveis

Existe muitas soluções de robôs colaborativos no mercado para diversas aplicações. Esses robôs são selecionados tendo em conta as seguintes características:

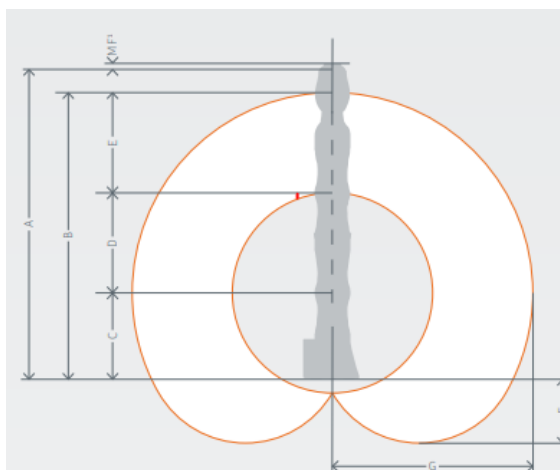
Carga útil: o peso que o robô colaborativo pode carregar. Todos os robôs colaborativos têm uma determinada carga útil, que é calculada sem o peso da ferramenta do robô.

Alcance: é a medida da distância que pode ser alcançada pelo pulso do robô colaborativo. Esta medição é feita em relação a base do robô.

Repetibilidade: é a proximidade de acordo entre várias posições alcançadas pelo robô colaborativo para a mesma posição controlada, repetida várias vezes nas mesmas condições.

Graus de liberdade (*Degrees Of Freedom - DOF*): é igual ao número total de deslocamentos independentes ou aspetos do movimento. Também é definido como a capacidade de movimento dos robôs.

O LBR iiwa 7 R800 (KUKA, 2014) representado na Fig. 6. foi desenvolvido pela KUKA. Este trabalha com precisão inflexível, processos que podem ser rastreados em detalhes e alta precisão, adequado para tarefas complexas de montagem e em situações de instalação confinada, por exemplo, este robô encontra-se instalado na fábrica do BMW Group em Dingolfing. Integra num espaço mínimo a montagem das caixas do diferencial para as transmissões do eixo dianteiro. E na Tabela 1. Apresenta a sua respetiva especificação técnica.



(a) Robô LBR iiwa 7 R800.

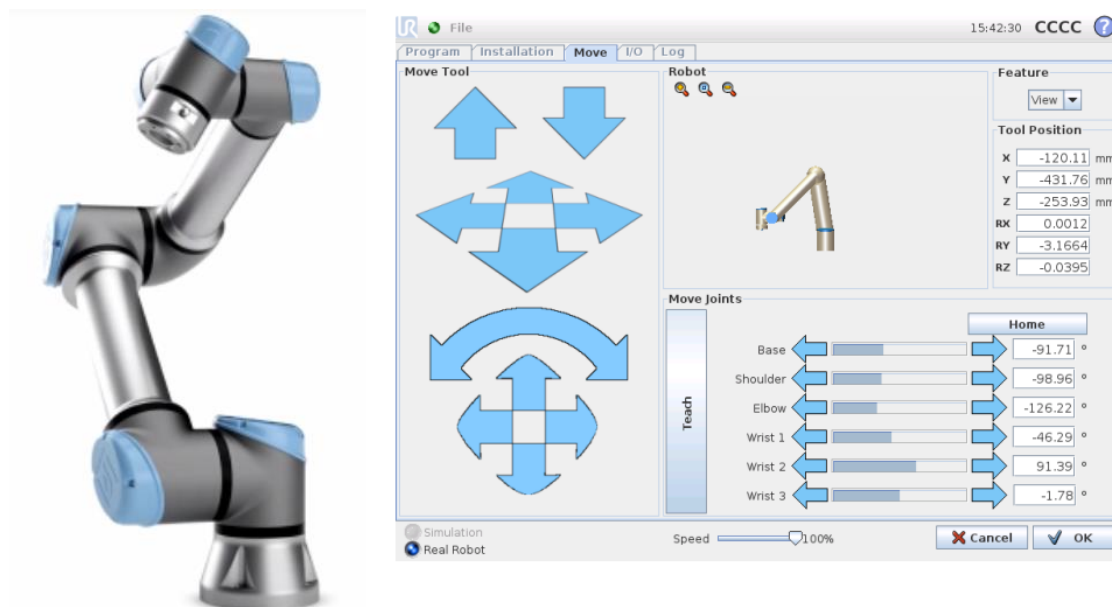
(b) Envelope de trabalho.

Fig. 6. Robô LBR iiwa 7 R800 e seu envelope de trabalho (KUKA, 2014).

Tabela 1. Especificação técnica do robô LBR iiwa 7 R800 (KUKA, 2014).

Carga útil [kg]	7	Alcance [°]	Binário max. [Nm]	Vel. max [°/s]	Dim. [mm]
Graus de liberdade	7	Eixo 1: ± 170	176	98	A: 1266
Repetibilidade [mm]	± 0.1	Eixo 2: ± 120	176	98	B: 1140
Posição de instalação	Qualquer	Eixo 3: ± 170	110	100	C: 340
Peso [kg]	23.90	Eixo 4: ± 120	110	130	D: 400
Índice de proteção	IP54	Eixo 5: ± 170	110	140	E: 400
Precisão do binário [%]	± 2	Eixo 6: ± 120	40	180	F: 260
Processador	Quad-core	Eixo 7: ± 175	40	180	G: 800
Disco rígido	SSD				
Interface	USB, Ethernet, DVI-I				
Índice de proteção	IP20				
Dim. control. [mm]	500x483x190				
Peso control. [kg]	23 kg				
Alimentação elétrica	110 a 230 V, 50/60 Hz				

O UR5 (Universal Robots, 2012), desenvolvido pela Universal Robots, é um robô industrial colaborativo leve e adaptável que lida com aplicações de serviço médio com flexibilidade máxima. e foi projetado para uma integração perfeita em uma ampla gama de aplicações, podendo observar as suas características técnicas na Tabela 2.



(a) Robô UR5e.

(b) Interface gráfico.

Fig. 7. Robô UR5e e interface gráfica para efetuar movimentos (Universal Robots, 2012).

Tabela 2. Especificações técnicas do Robô UR5e (Universal Robots, 2012).

Robô UR 5e	
Parâmetro	Valor
Carga útil [kg]	5
Graus de liberdade	6
Repetibilidade [mm]	± 0.03
Posição de instalação	Qualquer
Alcance [mm]	850
Potencia máxima [w]	570
Sensor força (alcance, precisão) [N]	50,3.5
Alcance de movimento por eixo [°]	± 360
Velocidade máxima por eixo [°/s]	± 180
Peso (incluindo cabo) [kg]	20,6
Índice de proteção	IP54
Materiais	Alumínio, plástico, aço
Porto I/O (digital in, out, analog in)	2, 2,2
Nível do ruído [dB]	<65
Temperatura de operação [°C]	0-50
Certificações	ISO 13849-1, PLd Category3, e ISO 10218-1

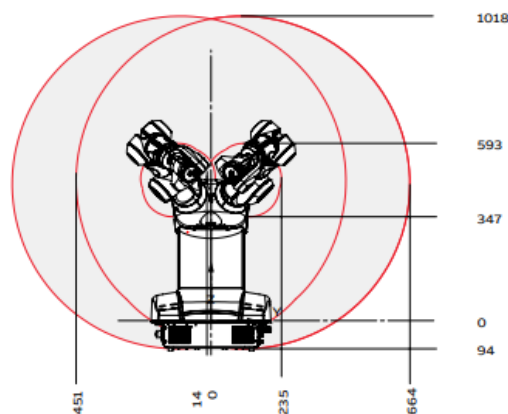
Tabela 3. Especificações técnicas do Robô UR5e (Universal Robots, 2012) (cont.).

Controlador	
Parâmetro	Valor
Índice de proteção	IP44
Temperatura de operação [°]	0-50
Porto I/O (digital in, out, analog in, out)	16, 16, 2, 2
Comunicação	Modbus TCP, ProfNet, Ethernet/IP, USB 2.0, 3.0
Fonte de alimentação	100-240VAC, 47-440 HZ
Tamanho [mm]	475x423x268
Peso[kg]	12
Consola	
Parâmetro	Valor
Programação	Ecrã tátil de 12" com polyscope interface
Índice de proteção	IP54
Peso[kg]	1.6
Comprimento do cabo [m]	4.5

Robô YuMi IRB 14000 (ABB, 2021) é o primeiro robô colaborativo com duplo arme. É capaz de trabalhar lado a lado nas mesmas tarefas que os humanos com extrema precisão, garantindo a segurança dos que estão ao seu redor. Foi projetado para atender às necessidades de produção ágil e flexível exigidas para a montagem de pequenas peças na indústria eletrônica. Também é adequado para outros ambientes de peças pequenas, incluindo a fabricação de relógios, brinquedos e componentes automotivos. Tudo isso graças aos seus braços duplos, mãos flexíveis, localização de peças baseada em câmara e controlo de movimento de última geração. Na Fig. 8 pode-se observar a imagem do robô Yumi. As suas respetivas características na Tabela 4.



(a) Robô ABB YuMi IRB 14000.



(b) Envelope de trabalho.

Fig. 8. Robô ABB YuMi IRB 14000 e seu envelope de trabalho (ABB, 2021).

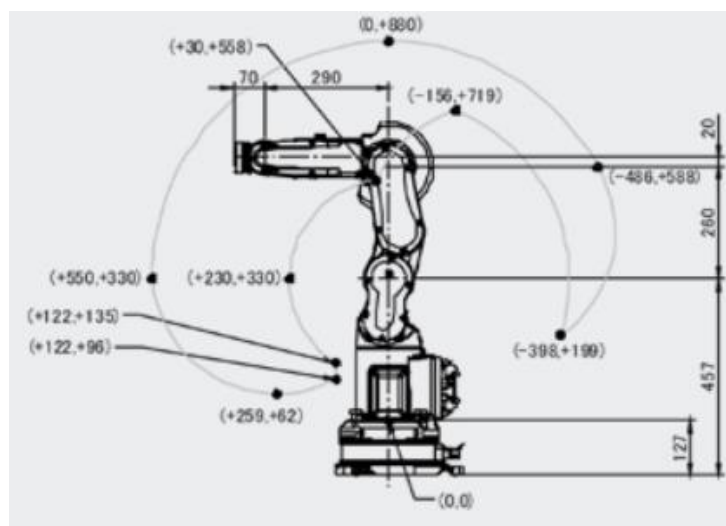
Tabela 4. Especificações técnicas do robô YuMi IRB 14000 (ABB, 2021)

Carga útil [kg]	0,5	Alcance [°]	Velocidade max. [°/s]
Graus de liberdade	14	Eixo 1: $\pm 168,5$	180
Repetibilidade [mm]	± 0.02	Eixo 2: -143,5 a 43,5	180
Alcance [mm]	559	Eixo 3: - 123,5 a 80	180
Controlador	integrado	Eixo 4: ± 290	400
Peso [kg]	38	Eixo 5: - 80 a 138	400
Índice de proteção	IP30	Eixo 6: ± 229	400
Temperatura de operação [°C]	5 a 45	Eixo 7: $\pm 168,5$	180
Certificações	PLB Cat B		
Velocidade máxima [m/s]	1,5		
Aceleração máxima [m/s*s]	11		

O Robot CR-4iA da FANUC (FANUC, 2020), apresentado na Fig. 9, é o menor na gama dos cobots, com design compacto e fino, aplicável em trabalhos leves, altamente manual e num espaço muito limitado. É possível ser montado na parede ou invertido, oferecendo uma ampla gama de movimento sem interferir no espaço de trabalho do operador. A Tabela 5 apresenta as respetivas especificações técnicas do robô CR-4iA.



(a) Robô FANUC CR-4iA.



(b) Envelope de trabalho.

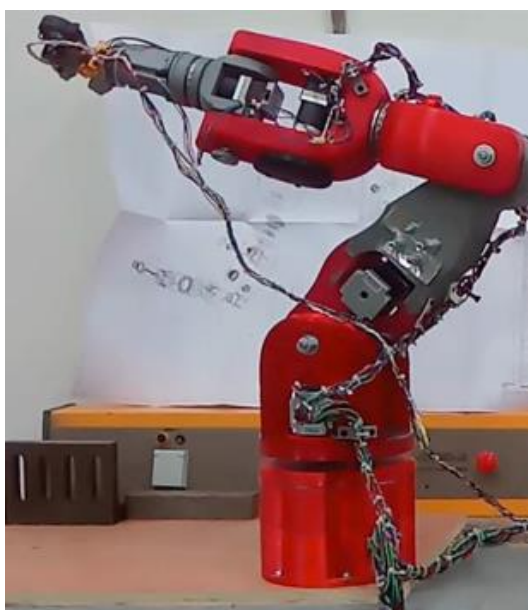
Fig. 9. Robô FANUC CR-4iA e seu envelope de trabalho (FANUC, 2020).

Tabela 5. Especificações técnicas do robô FANUC CR-4iA (FANUC, 2020).

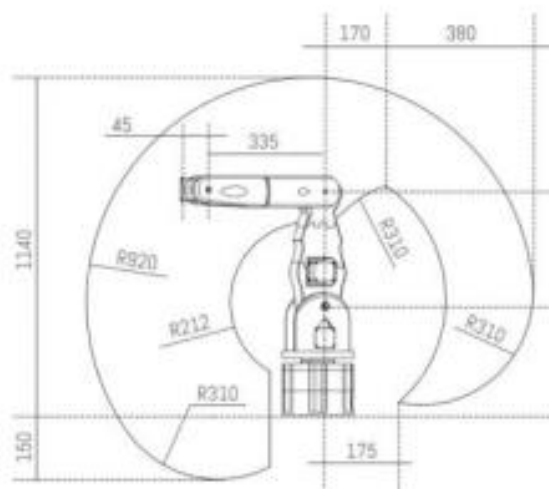
Carga útil [kg]	4	Alcance de movimento [°]	Momento/Inercia [Nm/kgm ²]
Graus de liberdade	6	Eixo1: 340	--
Repetibilidade [mm]	± 0.01	Eixo2: 150	--
Alcance [mm]	550	Eixo3: 354	--
Peso [kg]	48	Eixo4: 380	8.86/0.2
Velocidade máxima [mm/s]	1000	Eixo5: 200	8.86/0.2
Índice de Proteção	IP67	Eixo6: 720	4.9/0.067
Alimentação elétrica 50/60 Hz [V]	200-230		

2.4. Robôs baixos custos

O robô HydraX (Krimpenis *et al.*, 2020) apresentado na Fig. 10, trata-se de um braço robótico de baixo custo, impresso em impressora 3D. O robô foi projetado e fabricado sob medida em uma impressora 3D doméstica com PETG, de forma a atender tanto às propriedades funcionais quanto à estética. HydraX tem volume de trabalho suficiente para permitir manuseamento de peças médias e estabilidade para oferecer precisão.



(a) Robô HydraX.



(b) Envelope de trabalho.

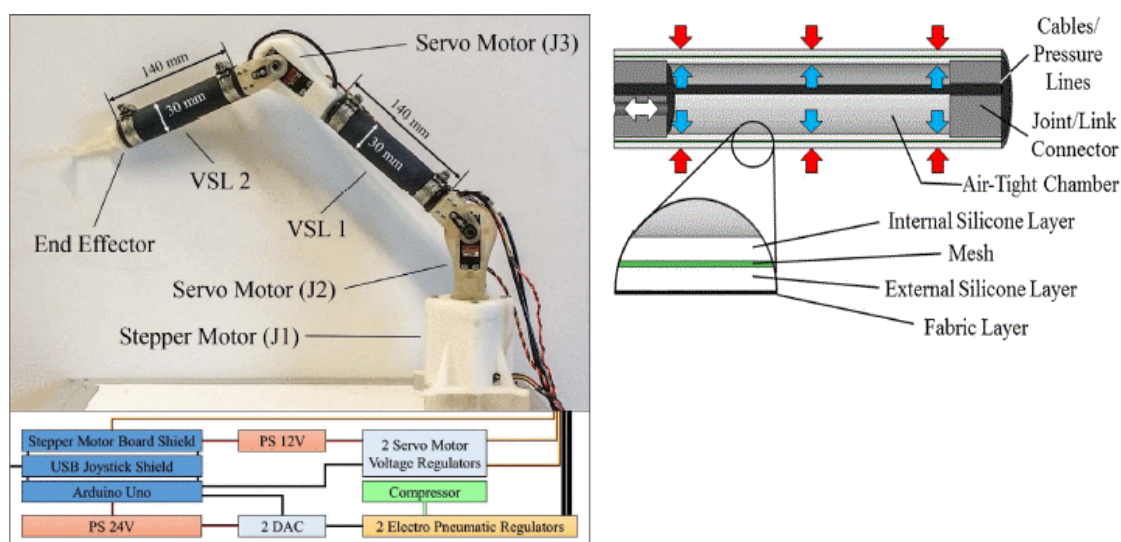
Fig. 10. Robô HydraX, dimensões e vista frontal do envelope do trabalho (Krimpenis *et al.*, 2020).

Como trata-se de um robô de baixo custo, os materiais utilizados foram de baixo custo e adaptáveis conforme representa na Tabela 6.

Tabela 6. Especificações técnicas do robô HydraX.

Carga útil [kg]	4	Alcance de movimento [°]	Momento/Inercia [Nm/kgm ²]
Graus de liberdade	6	Eixo1: 340	--
Repetibilidade [mm]	± 0.01	Eixo2: 150	--
Alcance [mm]	550	Eixo3: 354	--
Peso [kg]	48	Eixo4: 380	8.86/0.2
Velocidade máxima [mm/s]	1000	Eixo5: 200	8.86/0.2
Índice de Proteção	IP67	Eixo6: 720	4.9/0.067
Alimentação elétrica 50/60 Hz [V]	200-230		

O robô antropomórfico com rigidez variável (Stilli *et al.*, 2017) apresentado na Fig. 11, trata de um robô colaborativo com elo rígido variável (**Variable Stiffness Link - VSL**) e atua como sensor distribuído para detecção de colisão. O princípio de funcionamento do VSL está resumido na Fig. 11 (b), e é efetivamente uma câmara cilíndrica hermética feita de silicone encapsulada dentro de uma camada de tecido (reforça a expansão axial do VSL e evita a deformação do silicone). Além disso, uma malha de plástico é incorporada entre duas camadas de silicone. Ao aplicar pressão positiva a esta câmara faz com que o VSL varie sua rigidez. Para pressão alta, origina a força maior e, portanto, a maior rigidez do elo. Em baixa pressão, a malha plástica adiciona resistência estrutural e evita que o elo entre em colapso. Este robô é constituído por dois VSLs, três graus de liberdade, ambos rotacionais, e dois reguladores electropneumáticos (SMC ITV0030-3BS-Q, faixa de pressão de saída de 0 a 5 bar), são usados para controlar independentemente o nível de pressão dos dois VSLs. As articulações são acionadas por: um motor de passo de alto binário na base (SY57ST76/0686B, indicado como J1) e por servo motor (HS-7954SH, indicado como J2 e J3).



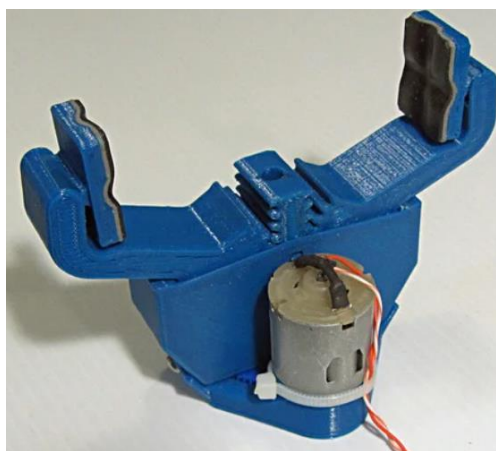
(a) Robô antropomórfico com rigidez variável. (b) Princípio de funcionamento do VSL.

Fig. 11. Robô antropomórfico com elo rígido variável e funcionamento de VSL (Stilli *et al.*, 2017).

O robô WE-R2.4 (Thingiverse, 2020) apresentado na Fig. 12, trata de um braço robótico de baixo custo, denominado como uma versão em escala de 80% do robô industrial UR3 da Universal Robots. Este resulta da combinação de seis atuadores empilhados para formar um braço robótico de 6 eixos. As estruturas do braço foram projetadas e impressas numa impressora 3D (MakerGear M2, com 0.2 mm de resolução, preenchimento de 20%, e com filamento PLA). Os movimentos das articulações (base, cotovelo e ombro) são feitas através do motor de passo NEMA 17 e as restantes articulações são acionada por motor de passo de ímã permanente (35PM048S8-08001) e atuação da garra é feita por motor DC (3VDC 4780 RPM).



(a) Robô WE-R2.4



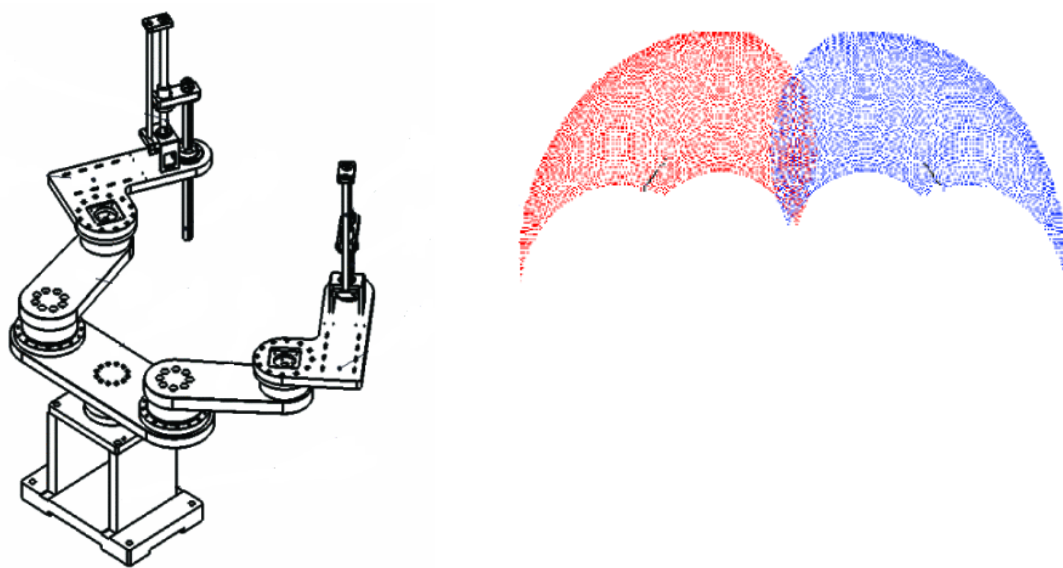
(a) Garra desenvolvida para WE-R2.4

Fig. 12. Robô WE-R2.4 e a garra (Thingiverse, 2020).

O robô SCARA com duplo braço (Liu *et al.*, 2018) apresentado na Fig. 13, foi desenvolvido com base em um robô SCARA de braço único para melhorar a adaptabilidade a tarefas complexas e a precisão na operação. Este foi projetado e analisado com base na simulação no MATLAB e na modelagem 3D do SOLIDWORKS.

O robô SCARA com braço duplo possui dois braços SCARA, simétricos, configurados de forma idêntica, e possui uma base giratória para expandir o alcance do robô. Os dois braços operacionais podem operar de forma independente e têm quatro graus de liberdade cada braço, sendo três rotacional e um de translação. A Fig. 13 mostra a estrutura do robô SCARA com duplo braço e o

espaço de trabalho quando a base não se move e na Tabela 6 mostra alguns dados e especificações do mesmo robô.



(a) Robô SCARA com duplo braço

(b) Envelope de trabalho sem o movimento da base

Fig. 13. Duplo robô SCARA e área do envelope do trabalho (Liu *et al.*, 2018).

Tabela 7. Especificações técnicas do robô SCARA com duplo braço (Liu *et al.*, 2018).

Nome	Elo	Alcance	Velocidade
Parâmetros	Cintura	360 [°]	200 [°/s]
	Braço superior	100 [°]	225 [°/s]
	Braço inferior	105 [°]	360 [°/s]
	Final	360 [°]	90 [°/s]
	Eixo Z	0 ≈ 200 [mm]	0.1 [m/s]
Repetibilidade [mm]	0,02		
Carga útil para cada braço [kg]	5		
Peso total do robô [kg]	90		

O robô humanoide leve (**Lightweight Humanoid - LWH**) (Yang *et al.*, 2019), imita o braço humano real em aspectos como formato e tamanho do membro, distribuição de massa e articulação, padrão de movimento e espaço de trabalho. Possui 8 graus de liberdade, com a estrutura e membros feitos em alumínio. A movimentação de cada eixo é feita através de servo motores controlados por Arduino. Este braço passou por muitos processos (projeto, simulação, construção e testes) e no final apresenta tamanho completo, leve, com capacidade de carga adequada e baixo custo, ou seja, tem 0,81 [m] de comprimento, 3,5 [kg] de peso e custa menos de

US \$ 1.000. Na Fig. 14 está representado a estrutura final do robô humanoide e a sua respetiva amplitude de movimento de todas as articulações na Tabela 7.

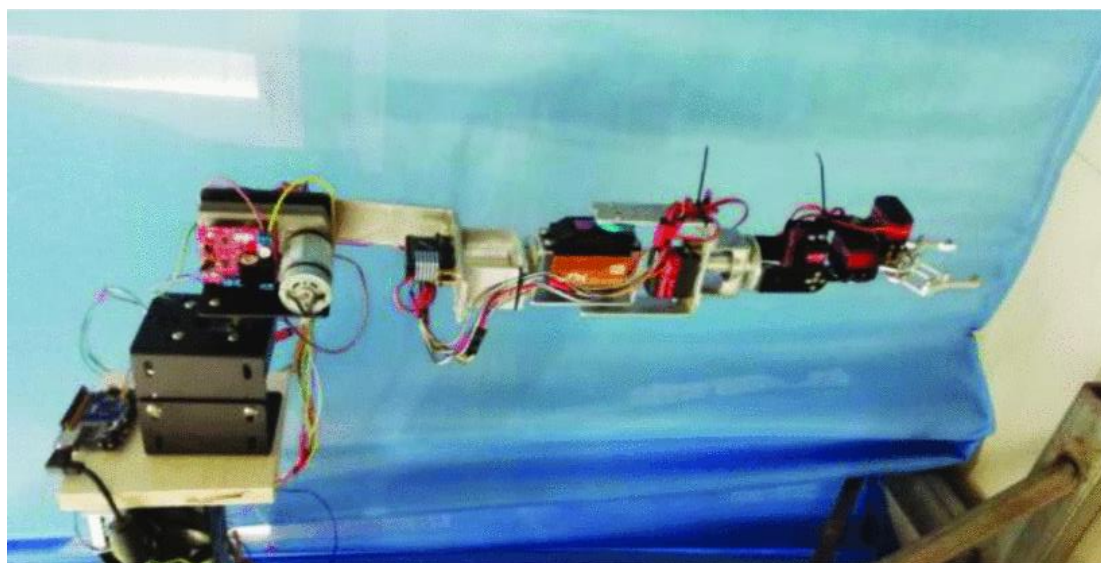


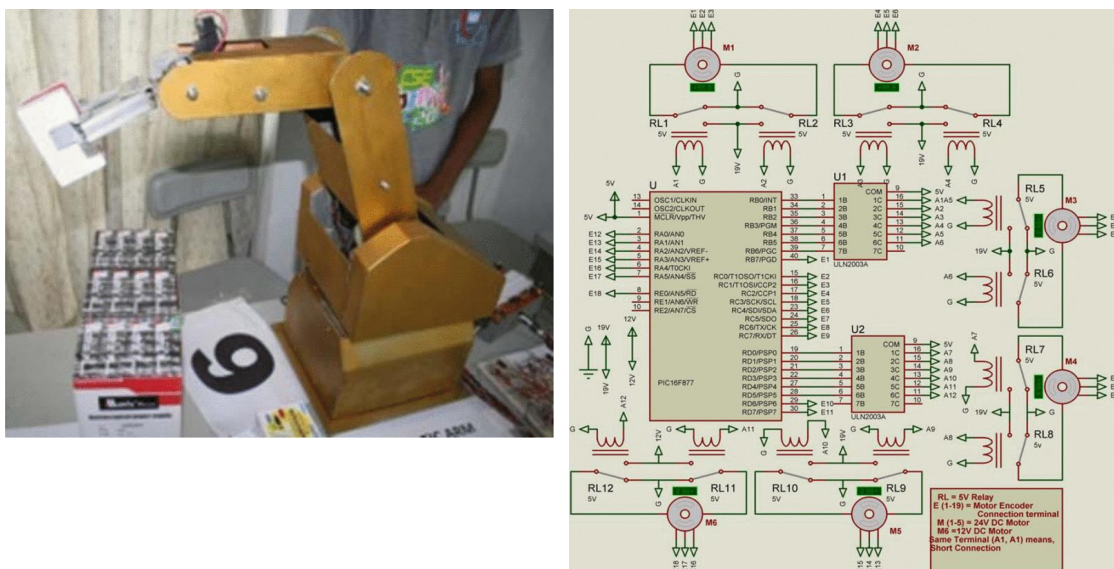
Fig. 14. Robô humanoide leve (Lightweight Humanoid - LWH) (Yang *et al.*, 2019).

Tabela 8. Amplitude dos movimentos do robô humanoide leve (Yang *et al.*, 2019).

Juntas	Eixo de rotação	Alcance [°]
Inclinação do ombro	Articulação do ombro para frente/trás	-180 a 50
Rotação do ombro	Rotação do ombro direita/esquerda	-45 a 180
Torção do braço	Torção da articulação do braço	-50 a 70
Rotação do cotovelo	Articulação do cotovelo	0 a 150
Rotação do antebraço	Articulação do antebraço	-90 a 90
Rolo de pulso	Articulação do pulso	-90 a 70
Inclinação do pulso	Articulação do pulso	-25 a 55
Rolo de mão(garra)	Abrir/fecha mão (garra)	0 a 60

O braço robótico com materiais recicláveis (Karmoker *et al.*, 2014) apresentado na Fig. 15, tem seis graus de liberdade, modelado com quatro elos em alumínio e tubo de aço inoxidável, sendo atuado através de motor de corrente contínua – (**Direct Current – DC**) conectado por correia dentada e engrenagem. Os motores são controlados através de microcontrolador um PIC (**Programmable Integrated Circuit**) com controlo de velocidade por modulação de pulso (**Pulse-Width Modulation – PWM**). O peso da estrutura completa de 7,363 [kg], é capaz de levantar até 1,5 [kg] e teve um custo final de \$US 125 incluindo materiais reciclados e novos. O

robô executa movimentos recebendo comando manual individual ou automático a partir da interface com o PC.



(a) Braço robótico com materiais recicláveis

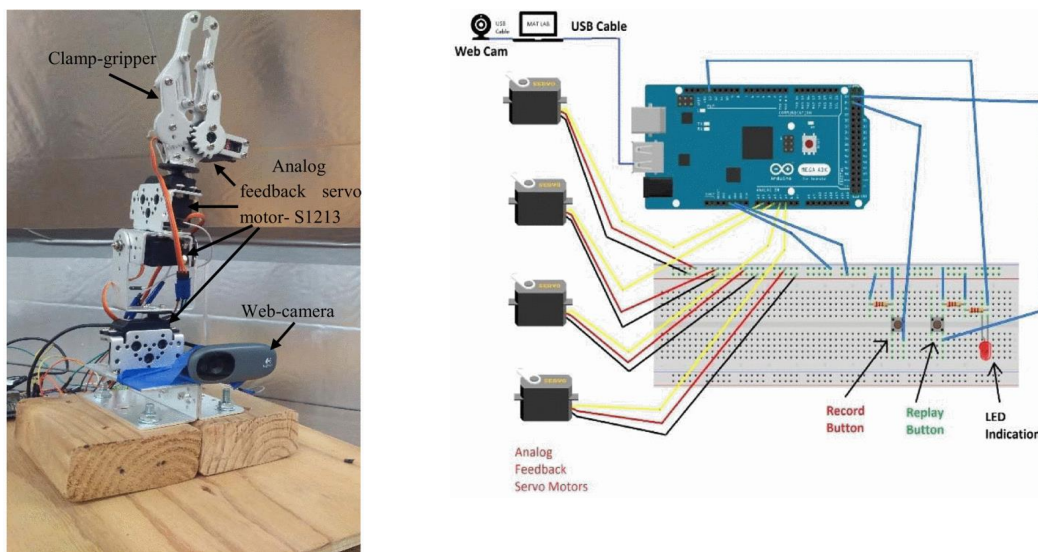
(b) Diagrama de circuito de controlo do braço robótico

Fig. 15. Braço robótico com materiais recicláveis e diagrama de circuito de controlo (Karmoker *et al.*, 2014).

O braço robótico com orientação baseada em visão computacional (Sundaram *et al.*, 2019) apresentado na Fig. 16, tem 6 DOF, ao qual se anexa uma garra. Este é controlado através do microcontrolador Arduino Mega 2560. A orientação baseia na visão computacional, que utiliza uma câmara de baixo custo para detetar o(s) objeto(s) em que irá pegar. Esta também auxilia em situações em que o objeto a ser manipulado é deslocado do local que o braço foi originalmente treinado para pegar. A estrutura do braço é em alumínio e movimento é conseguido através dos servos motores. A câmara está instalada na base do robô e conectada ao PC. Os dados da câmara são processados no software MATLAB, através de um algoritmo de deteção de objetos. Em seguida, é comunicado ao Arduino a distância entre o braço e objeto.

Foram feitos vários testes e com base na Tabela 8 conclui-se que, se o objeto for deslocado 1 [cm], o braço robótico treinável falha ao pegar o objeto, enquanto o braço robótico baseado com deteção de objetos baseada em visão computacional deteta, localiza e pega com sucesso o objeto deslocado. Também pode ser observado que o braço robótico com deteção de objetos baseada em

visão computacional falha em detetar e localizar um objeto se o deslocamento deste for superior a 12 [cm].



(a) Estrutura de braço robótico baseada em visão

(b) Diagrama do circuito de controlo

Fig. 16. Estrutura de braço robótico com deteção baseada em visão computacional e seu circuito de controlo (Sundaram *et al.*, 2019).

Tabela 9. Ensaio efetuado com o braço robótico com deteção baseada em visão computacional (Sundaram *et al.*, 2019).

Número de teste	Deslocamento em relação a posição original [cm]	Objeto agarrado pelo braço	
		sem auxílio a visão	com auxílio a visão
1	1,00	Não	Sim
2	3,00	Não	Sim
3	5,00	Não	Sim
4	7,00	Não	Sim
5	10,00	Não	Sim
6	11,00	Não	Sim
7	12,00	Não	Sim
8	12,50	Não	Não
9	13,00	Não	Não
10	13,50	Não	Não

O braço robótico macio (*Soft Robotic Arm - SRA*), (Arm *et al.*, 2018) apresentado na Fig. 17, é feito de tecido de nylon revestido com poliuretano termoplástico (*Thermoplastic*

polyurethane - TPU). Este contém quatro articulações que podem dobrar verticalmente e horizontalmente e uma pinça para pegar objetos de diferentes formas e tamanhos. Trata de um braço de baixo custo, leve, com peso total de 2 [kg], 1 [m] de comprimento e 6,6 [cm] de diâmetro que permite uma interação segura com operador.

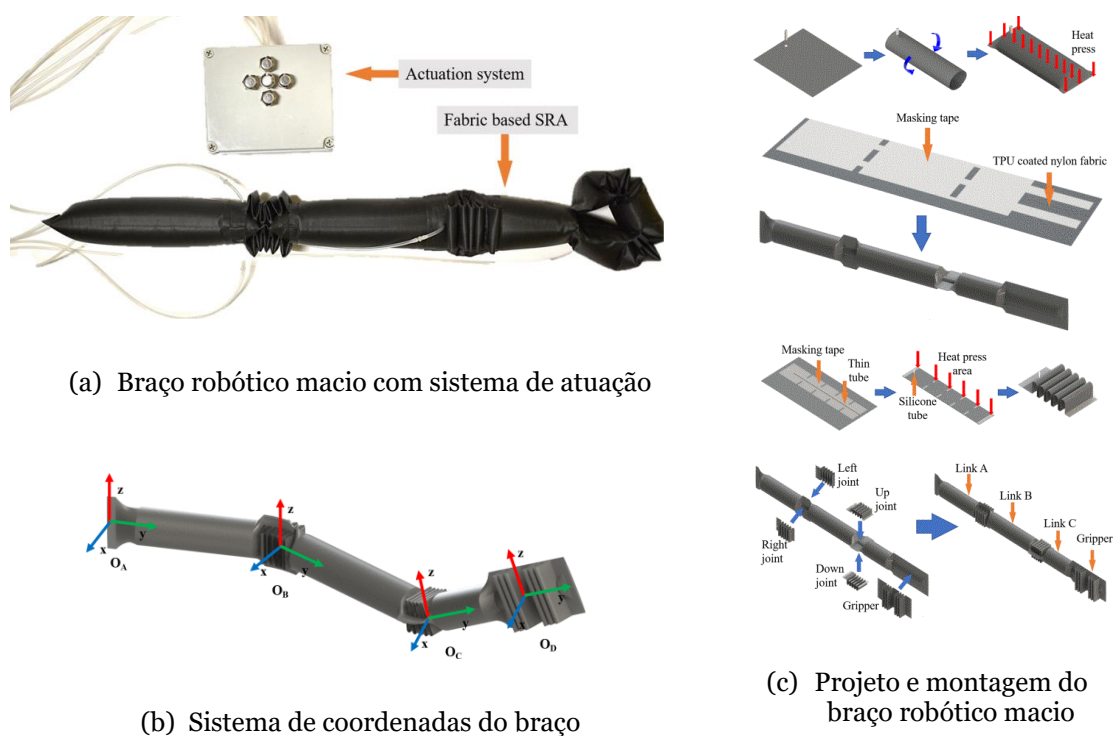
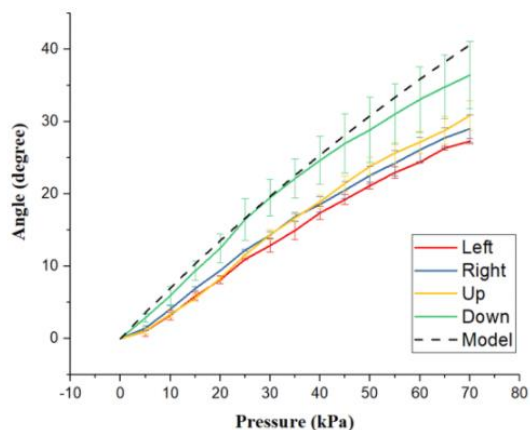


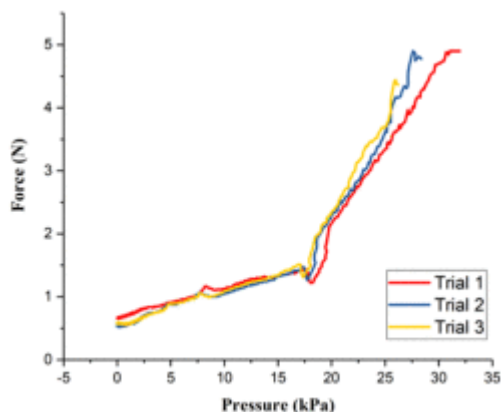
Fig. 17. Estrutura, projeto e montagem do braço robótico macio (Arm *et al.*, 2018).

Os ângulos de limite para a esquerda, direita, cima e para baixo são: $27,3^\circ \pm 0,36$, $29,03^\circ \pm 1,89$, $30,87^\circ \pm 2,04$, $36,43^\circ \pm 4,65$ respectivamente, sendo apresentada na Fig. 18 (a) a variação do ângulo em função da pressão e na Fig. 18 (b) a força que braço suporta com a variação de pressão.

O braço robótico contém seis atuações (corpo principal, quatro atuadores de junta e a garra), sendo todas as atuações realizadas por ar comprimido nos atuadores. As atuações são controladas por electroválvulas e sensores de pressão para fornecer feedback da pressão. Este sistema é controlado por microcontrolador Arduíno Mega 2560 conforme representado na Fig. 19.

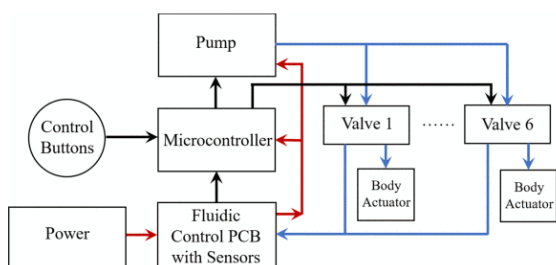


(a) Resultado do ângulo de curvatura sob pressão.

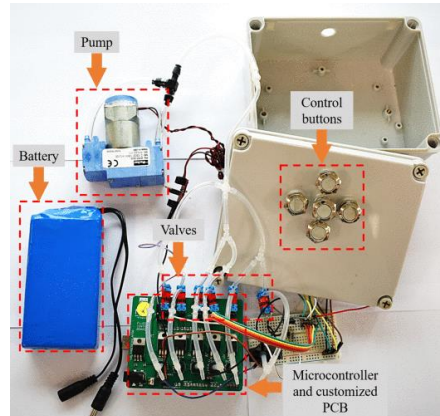


(b) Resultado da força suportada sob pressão.

Fig. 18. Resultado do ângulo de curvatura e a força suportada sob pressão do braço robótico macio (Arm *et al.*, 2018).



(a) Fluxograma do sistema de controle e atuação.



(b) Protótipo real do sistema de controle.

Fig. 19. Fluxograma e protótipo real do sistema de controle e atuação do braço robótico macio (Arm *et al.*, 2018).

2.5. Nota conclusiva

Ao longo deste capítulo foram apresentadas várias configurações dos robôs colaborativos. Os robôs colaborativos são amplamente usados atualmente e sua aplicação continua em crescimento devido a sua interação com o ser humano. No entanto, continua a ser um grande desafio pesquisar e desenvolver um braço robótico devido a sua complexidade. Existe muitas questões pendentes a serem determinadas antes de uma implementação real, envolvendo forma e tamanho dos elos, peso, densidade e rigidez do material, modo de acionamento e binário dos motores, amplitude e padrões de movimento, capacidade de carga, etc.

Os braços robóticos tradicionais, são feitos de materiais rígidos, são usados principalmente para manipular e montar itens em ambiente industrial. Possuem como vantagens a elevada produção de força, controlo de alta precisão e alta eficiência. No entanto, quando se trata de interações humano-robô, a segurança é sempre a principal preocupação, o que significa que os braços robóticos tradicionais podem não ser uma escolha adequada devido à sua rigidez. Além disso, o elevado custo e o grande peso também dificultam o desenvolvimento e a aplicação de braços robóticos tradicionais.

Os robôs colaborativos comerciais apresentam parâmetros características técnicas que aliam as vantagens dos robôs industriais tradicionais, como sejam a elevada repetibilidade (poderá chegar a 0.01 [mm]), carga útil media, e número médio de graus de liberdade (seis a sete graus de liberdade) e robustez, à capacidade de salvaguardar e promover a interação humano-robô. Embora neste momento existam muitas soluções fiáveis nos mercados, muitas otimizações estão a ser realizadas para garantir a segurança e estabilidade dos robôs colaborativos, que se refletem no elevado custo de aquisição.

Os robôs colaborativos de baixos custos, têm o propósito em alcançar os mesmos objetivos que os restantes, mas com baixo custo de construção, fiabilidade e garantia da segurança. Nas soluções apresentadas as estruturas tendem sempre a ser mais leve possível, daí serem impressos com filamento PLA ou com estrutura em alumínio. Neste tipo de estrutura deve-se aproveitar o máximo do binário do motor para carga útil e também para reduzir os danos ao humano durante a colisão. Grande parte dessas soluções de baixos custos utilizam motores de passo devido elevado binário e elevada precisão no posicionamento. Quanto ao controlo, a maioria das vezes são controlados por microcontroladores.

3. Materiais e Métodos

Foi efetuada revisão bibliográfica das soluções dos robôs comercializáveis, bem como de soluções de braços robóticos de baixos custos. Os materiais e métodos foram selecionados com referência aos artigos científicos e também com base em alguns projetos existente na literatura.

3.1. Materiais

Neste subcapítulo apresentamos os materiais subdividindo como hardware e software. Como hardware, apresentamos alguns materiais e processos que faz parte física da estrutura do braço desenvolvido e como software apresentamos todos os programas utilizados no desenvolvimento do mesmo.

3.1.1. Hardware

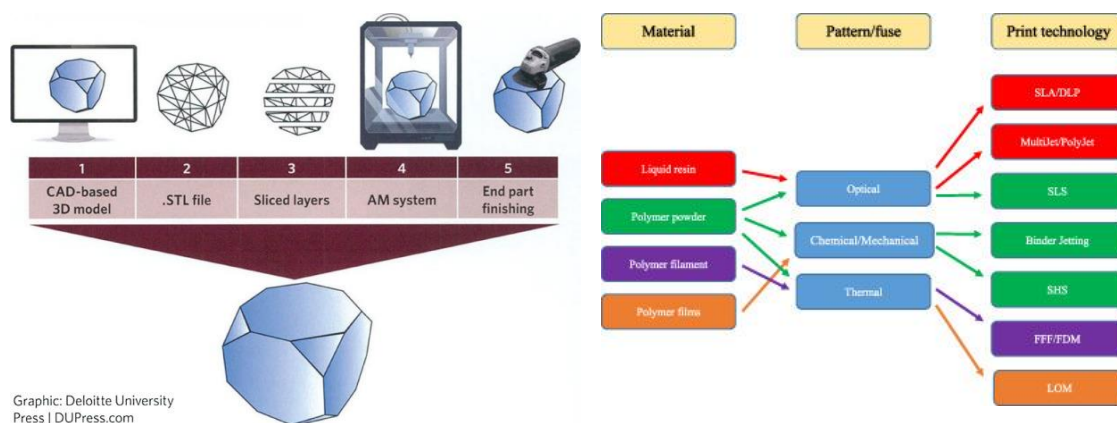
Nesta seção, abordamos temas como impressão e impressora 3D, filamento termoplástico, controlo e movimento (motores, encoder e controlador do motor), microcontrolador (Esp32), protocolos de comunicação (Esp Now, UART, I2C).

3.1.1.1. Impressão 3D

A impressão 3D (Dizon *et al.*, 2018), conhecida como manufatura Aditiva (AM), já está sendo adotada para prototipagem e fabricação rápida. Recentemente, foram desenvolvidas técnicas de AM mais baratas e mais rápidas que podem produzir elevadas qualidades de impressão. A Fig. 20 (a), apresenta os processos da impressão 3D, começando por desenho de modelo de projeto auxiliado por computador através de software CAD, que posteriormente guardado em formato de arquivo STL (**Standard Triangle Language – linguagem de estereolitografia ou tesselação de estereolitografia**), impressão do desenho e por último acabamento final.

Para muitas aplicações, os processos subtrativos impõem uma série de restrições, incluindo as ferramentas dispendiosas, acessórios e a necessidade de montagem de peças complexas. Além disso, os processos de fabricação subtrativos, como a maquinação, podem resultar no desperdício de até 90% do bloco original de material.

Na literatura existem vários tipos de impressão 3D (Stansbury & Idacavage, 2016), como: Modelação por deposição de material fundido (*Fused Deposition Modeling – FDM*), Estereolitografia (*Stereolithography – SLA*), Processamento de Luz Direta (*Digital Light Processing – DLP*), Sinterização Seletiva Laser (*Selective Laser Sintering – SLS*), Sinterização Direta Laser de Metal (*Direct Metal Laser Sintering – DMLS*), Fusão Seletiva a Laser (*Selective Laser Melting – SLM*), Feixe de eletrões (*Electron Beam Melting – EBM*), e Fabricação de Objetos Laminados (*Laminated Object Manufacturing – LOM*). A Fig. 20 (b) apresenta uma visão de métodos de fabricação de aditivos para diferentes materiais(polímero).



(a) Processos para impressão 3D.

(b) Tipos de materiais, mecanismos e métodos de fabricação aditivos.

Fig. 20. Processos para impressão 3D e visão dos tipos de materiais e métodos de fabricação aditivas (Dizon *et al.*, 2018).

Modelação por deposição de material fundido (Dizon *et al.*, 2018) é um dos processos de fabricação aditiva amplamente utilizados para fabricar protótipos e peças funcionais em plásticos de engenharia comuns. O processo é baseado na extrusão de filamentos termoplásticos, sendo estes aquecidos através de uma extrusora e, através de um bico, depositados em camadas numa plataforma para construir peças, camada por camada, a partir de um modelo digital da peça. A simplicidade, confiabilidade e acessibilidade do processo FDM tornaram a tecnologia de fabricação aditiva amplamente reconhecida e adotada pela indústria, pelo meio académico e pelos consumidores.

Uma impressora 3D para modelação de deposição de material fundido, conforme exemplo apresentado na Fig. 12 (ElectroPark, 2020), é um equipamento eletromecânico capaz de executar movimentos até 3 ou 4 eixos. Ao executar esses movimentos e derretendo os filamentos, origina o desenho 3D físico. Normalmente estas impressoras são constituídas por: base de impressão,

extrusora (componente que derrete o filamento), sistema de porta filamento, motores para movimentação cartesiana (x, y, z) e um controlado.

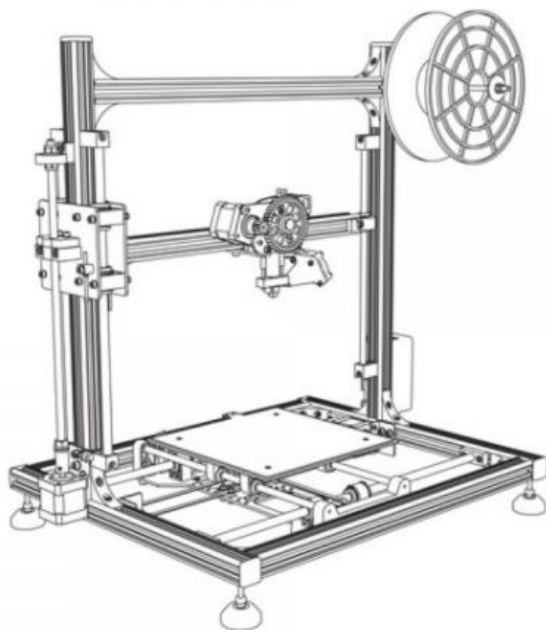


Fig. 21. Representação da impressora 3D.

Filamento termoplástico é o material utilizado para extrusão no processo da modelagem de deposição fundida. Por meio do processo FDM, ambos são derretidos e, em seguida, extrudidos por meio de um bico para formar as camadas que criam a peça final.

Existe vários tipos filamentos, os mais comuns utilizado na impressão 3D (All3DP, 2021) são: ácido polilático (*Polylactic acid* – PLA), acrilonitrila butadieno estireno (*Acrylonitrile butadiene styrene* – ABS), polietileno tereftalato glicol (*Polyethylene Terephthalate Glycol* – PETG). A Tabela 9 contem alguns parâmetros e propriedades dos filamentos mais utilizados.

Tabela 10. Propriedades dos filamentos mais utilizados na impressão 3D.

Propriedades	Filamentos		
	PLA	ABS	PETG
Temperatura de transição [°C]	60	105	80
Densidade [g/cm ³]	1,3	1,0 – 1,4	1,27
Temperaturas de extrusão [°C]	180 – 210	230 – 250	220 – 250
Biodegradabilidade	Sim	Não, reciclável	Não, reciclável
Aquecimento da base, temperatura [°C]	--, --	Sim, 90 – 110	Sim, 90 – 100

PLA, tem maior dureza superficial, é o mais difícil de dar acabamento se necessário, produz os melhores resultados com o resfriamento adequado, baixo encolhimento térmico e ausência de odor durante a impressão. Ideal para impressões 3D onde a estética é importante. Devido à sua temperatura de impressão mais baixa, é mais fácil de imprimir e, portanto, mais adequada para peças com detalhes finos, produz os melhores resultados com o arrefecimento adequado (Hubs, 2021; All3DP, 2021).

ABS, é mais difícil imprimir, mas mais adequado para aplicações onde resistência, ductilidade, facilidade em cortar, tornear, fresar e cortar e estabilidade térmica são requeridas. Está sujeito ao empenamento, geralmente é inodoro no processo de impressão. Necessita ser impresso numa impressora 3D fechada, para que arrefeça muito lentamente (Hubs, 2021).

PETG, é um material resistente, extremamente durável, não é quebradiço, imprime sem odor, com um encolhimento muito baixo, sem empenamento, ideal para imprimir grandes peças, grande adesão da camada (rigid.ink, 2021).

A impressão da estrutura do braço robótico proposto foi realizada com filamento de PLA devido às suas características, propriedades e bons resultados alcançado por outros na literatura em desenvolver estruturas de braços robóticos com filamento de PLA.

3.1.1.2. Controlo e movimento

Os motores elétricos são dispositivos elétricos que transformam energia elétrica em mecânica, sendo constituídos por um rotor e por um estator. Os motores estão divididos em duas classes, motores de corrente alternada e contínua. Existem vários tipos de motores de corrente contínua. Motores que possuem enrolamento de campo ou ímã permanente e podem ter ou não escovas. Estes motores são acionados a partir de uma fonte de corrente contínua, possuem diversas aplicações e são destacados pelo fácil controlo e a robustez.

Um **motor de passo** é um transdutor eletromagnético incremental que permite uma conversão de impulso elétricos em deslocamentos angulares. A cada impulso corresponde um deslocamento elementar (rotativo ou linear), designado por passo e a frequência desses impulsos impõe a frequência angular. Estes motores têm uma grande importância e vantagens nas aplicações robóticas devido a resolução no posicionamento, seu tamanho, forma de controlo e grande valor binário (Krimpenis et al., 2020).

Encoder rotacional são dispositivos eletromecânico que conta ou reproduz pulsos elétricos a partir do movimento rotacional de seu eixo. São utilizados para conversão de movimentos rotativos ou deslocamentos lineares em impulsos elétricos de onda quadrada. Este dispositivo permite calcular o deslocamento, velocidade, sentido de movimento, e a posição atual.

Existe 2 tipos de encoder rotacional sendo, incremental e absoluto. Estes são diferenciados por a posição do encoder incremental é dada através de pulso, enquanto a posição do encoder absoluto é determinado pela leitura de um código e exclusivo para cada posição, está posição nunca é perdível com a falha de energia. Estes sinais são originados por movimento do disco. Na Fig. 22 está representado o diagrama do disco codificador para os encoder (Electronics Hub, 2019).

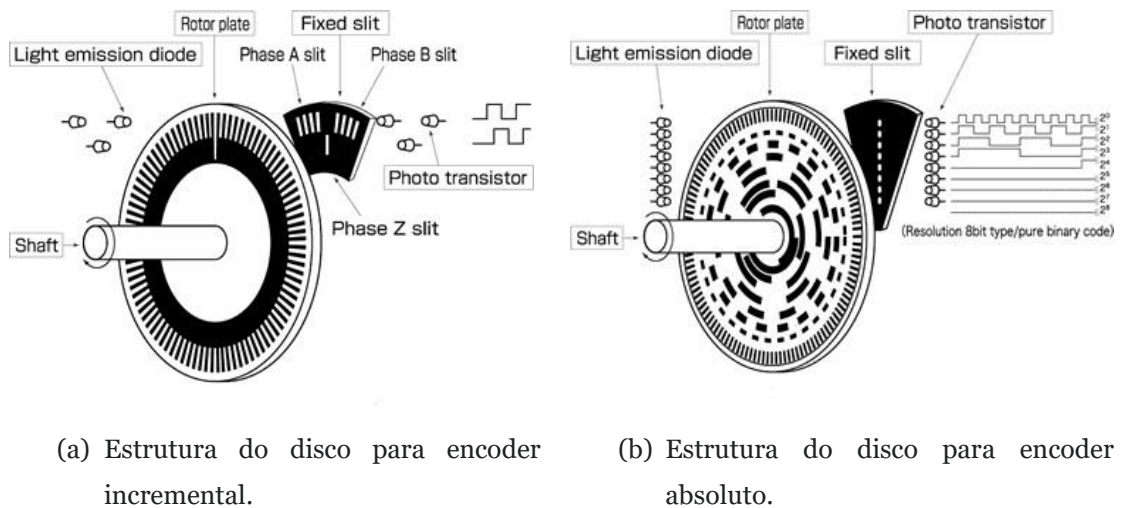


Fig. 22. Diagrama do disco codificador dos encoder.

3.1.1.3. Microcontrolador

Os microcontroladores fazem parte do nosso dia-a-dia, seja num carro, computador, telemóvel, tablet, máquinas industriais, sistemas inteligentes, domótica, robótica, etc. Os microcontroladores são grande parte do mundo moderno (Arrow.com, 2020).

Um microcontrolador é um microcomputador fabricado a partir da integração em larga escala (*Very large-scale integration – VLSI*) sendo, VLSI o processo de criação de um circuito integrado combinando milhões de transístores MOS (Metal-Oxide-Semiconductor), normalmente de silício num único chip. Trata-se de um microcomputador compacto fabricado para controlar as funções de sistemas embutidos, normalmente usado para aplicações específicas para determinadas tarefas. O nome microcontrolador advém pelo facto de em único circuito integrado conter: Unidade central de processamento (*Central Processing Unit – CPU*), memórias, temporizadores/contadores, conversores analógico-digital (ADC – *Analog-to-Digital Converter*) /digital-analógico (DAC – *Digital-to-Analog Converter*), periféricos de entrada/saída, barramento de dados e endereços.

Microcontrolador ESP 32, é uma série de microcontrolador SoC (System on Chip) de baixo custo e baixo consumo energético, projetada e desenvolvida pela empresa Espressif Systems. Trata-se de um microcontrolador com diversos recursos e uma variedade de periféricos integrados para uma ampla gama de aplicações. A série ESP32 emprega dois núcleos, CPU Tensilica Xtensa LX6 de 32 bits, 448 KB de memória ROM (*Read-Only Memory*), 520 KB memória RAM (*Random Access Memory*), 4 MB memória flash, placa de comunicação Wireless padrão 802.11 b/g/n, conexão Wifi de 2.4GHz, antena embutida na placa, modos de operação STA/AP/STA+AP, Bluetooth BLE 4.2, interface periférica (GPIO, ADC, sensor Hall, DAC, UART, I2C, I2S, SPI, PWM) e pode funcionar individualmente a frequência até 240 MHz, (Babiuch *et al.*, 2019). A Fig. 23 apresenta o diagrama de bloco de função do ESP32 como todos os recursos existentes.

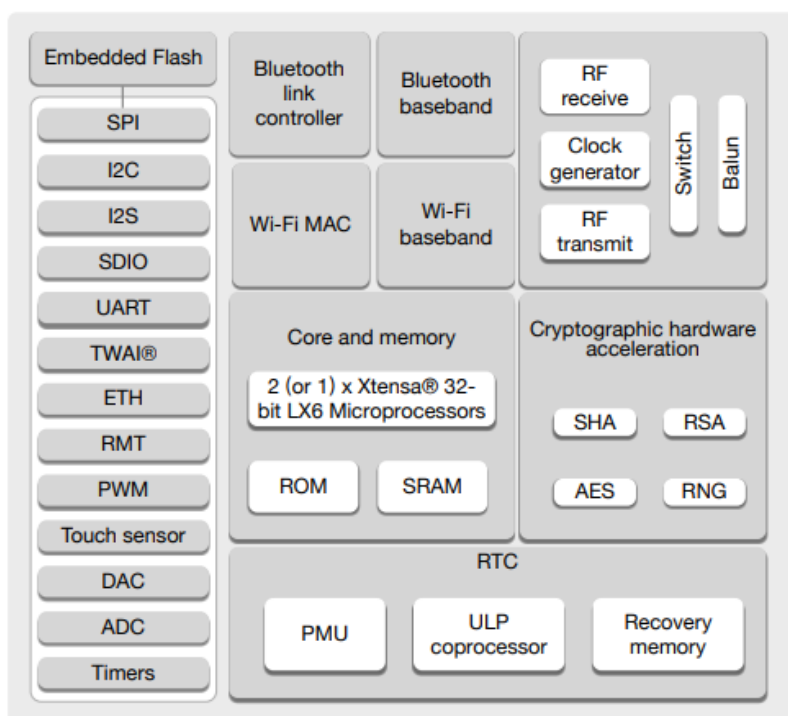


Fig. 23. Diagrama de bloco de função do ESP2 (Espressif, 2021).

Devido a compatibilidade e recursos, este microcontrolador está em destaque em diversos projetos. Por exemplo, Biswas & Iqbal (2018) propõem monitorização do sistema para bombeamento solar automatizado de baixo custo para irrigação em países em desenvolvimento. Bunkum *et al.* (2019) propõem o desenvolvimento de braço robótico para uso em sala de cirurgia, que é controlado por joystick de qualquer lugar. Bolla *et al.* (2020) propõem a implementação de um posto de combustível inteligente no processo de abastecimento usando a tecnologia da internet das coisas (*Internet of Things – IoT*) dado que a maioria dos postos de combustível atualmente são operados manualmente, o que leva atraso e exige mais mão de obra.

Protocolos de comunicações, são utilizados em diversas áreas tecnológicas, possibilitando a troca de informações entre diferentes dispositivos, e estabelecendo regras e convenções que regem o funcionamento de diferentes comunicações. Os protocolos de comunicação com fio como UART (*Universal Asynchronous Receiver Transmitter*), SPI (*Serial Peripheral Interface*) e I2C (*Inter-Integrated Circuit*) são as interfaces de hardware comuns nos microcontroladores (RoboCore, 2021). No ESP32, para além de protocolos de comunicação comuns (UART, SPI, I2C), encontram-se disponíveis os, I2S (*Inter-IC Sound*), Ethernet MAC interface, e CAN (*Controller Area Network*). Para além dos protocolos de comunicação com fios, estão disponíveis no ESP32, protocolos de comunicação sem fios como, WiFi, Bluetooth e ESP-NOW (Espressif, 2021).

UART (Fang & Chen, 2011), trata-se de um sistema de comunicação série assíncrona, ou seja, sequências de bit. Utiliza apenas duas linhas de sinal (RX, TX) para completar a comunicação *full-duplex*. Para estabelecer a comunicação é necessária uma sequência de bit constituído por: bit de início, bit de dados, bit de paridade e bit de paragem. O bit de início (estado lógico 0) é usado para alertar o recetor do início da comunicação e para forçar o relógio do recetor a sincronizar com o relógio do transmissor. Os bits de dados normalmente são 8 bits e são enviados um de cada vez à mesma velocidade de transmissão. Quando todos os bits de dados forem transmitidos, o transmissor pode adicionar um bit de paridade (par ou ímpar) para o recetor detetar eventuais erros na comunicação. Posteriormente é enviado o bit de paragem (estado lógico 1) para finalizar a comunicação. Os bits são enviados/recebidos a uma dada cadência acordada entre emissor (TX) e recetor (RX) a que se chama taxa de transmissão (BR – *baud rate*), medida em bits por segundo. Quando não existe a transmissão de dados, a linha de transmissão fica ativa (estado lógico 1). A Fig. 24 apresenta o diagrama e estrutura de ligação da comunicação UART.

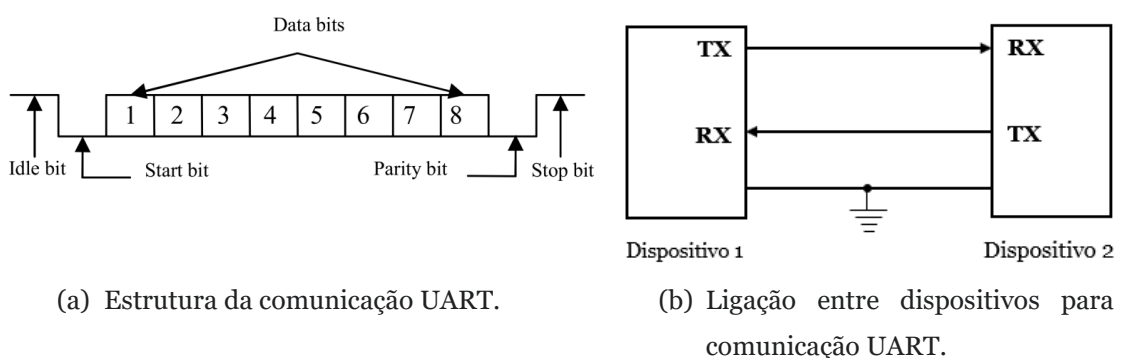
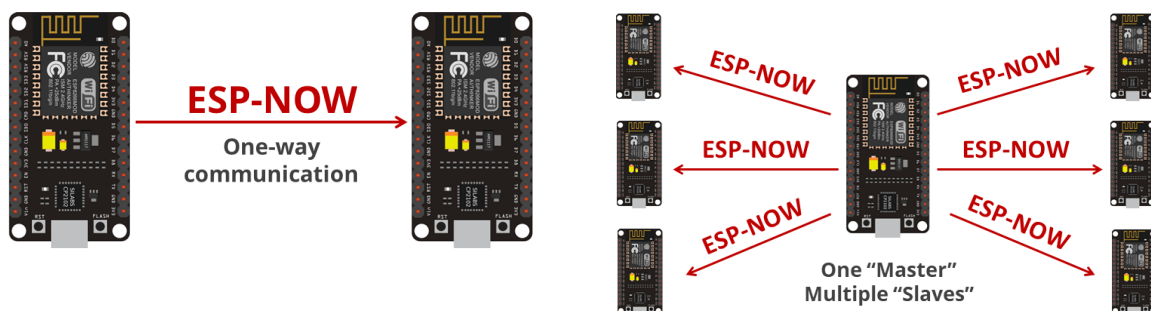


Fig. 24. Estrutura e ligação para comunicação UART.

O protocolo de comunicação **ESP-NOW** (Espressif, 2021) é um protocolo de comunicação rápido que pode ser usado para trocar pequenas mensagens. Foi desenvolvido pela **Espressif** e permite

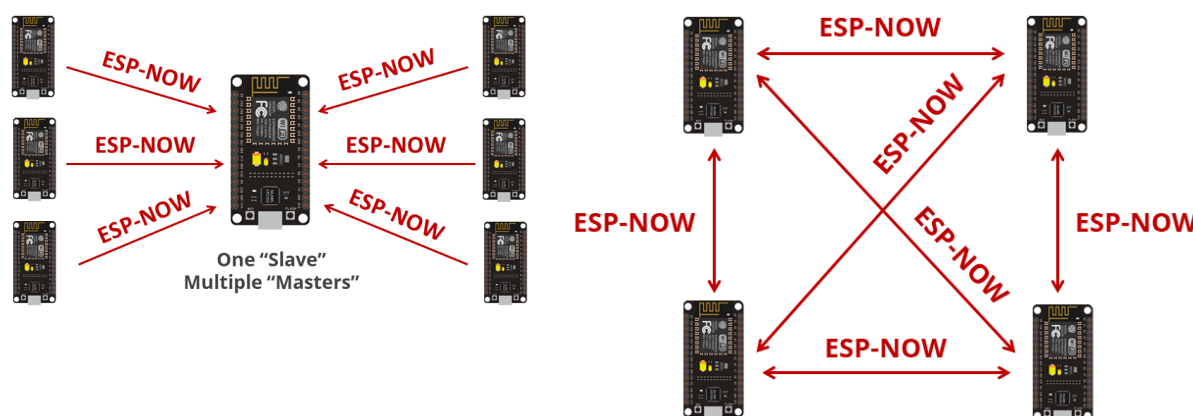
que vários dispositivos se comuniquem sem usar WiFi. O protocolo é semelhante à conectividade sem fio de 2,4 GHz de baixa potência que geralmente é implantada em ratos sem fio e tem uma taxa de transmissão de 1 Mbps. O ESP-NOW aplica a tecnologia **IEEE 802.11**, juntamente com a função **IE** desenvolvida pela Espressif e a tecnologia de criptografia (CCMP – *Counter Mode with Cipher Block Chaining Message Authentication Code Protocol*), para proporcionar solução de comunicação segura e sem conexão WIFI.

O protocolo ESP-NOW apresenta alguns recursos como: comunicação unicast criptografada e não criptografada, pode transportar até 250 bytes, função de retorno que pode ser definida para informar a camada de aplicativo sobre o sucesso ou falha da transmissão, não suporta *Broadcast*, no modo estação pode suportar até 10 dispositivos pares criptografados e até 20 pares não criptografados incluindo pares criptografados. Para estabelecer a comunicação precisa é necessário fazer um registo em ambos os dispositivos com os seguintes dados: endereço de controlo de acesso ao meio (MAC – *Media Access Control*), número de canal (0 -15) e informação se os dados são ou não criptografada. Este protocolo de comunicação é muito versátil e permite comunicação unilateral ou bidirecional em diferentes configurações como apresentado na Fig. 25.



a) Comunicação unilateral.

b) Comunicação unilateral entre um master e vários slaves.



c) Comunicação unilateral entre vários masters e um slave.

d) Comunicação bilateral entre vários ESP32.

Fig. 25. Diferentes configurações de comunicação com o protocolo ESP NOW (Gnd_To_Vcc, 2021).

I2C (Bagdalkar & Ali, 2020), trata-se de um protocolo de comunicação síncrono com interface a dois fios (SDA – *serial data* e SCL – *serial clock*) para estabelecer comunicação série síncrona entre um mestre e um ou mais dispositivo escravo. Para estabelecer a comunicação, o dispositivo gera o sinal de relógio para as sequências de bits como: um bit para condição do início da comunicação, sete bits de endereço, um bit para leitura/escrita (R/\bar{W}), dois bits ACK/NACK – *acknowledge/not acknowledge* (0/1) e um bit para fim da comunicação. A Fig. 26 representa a estrutura da comunicação I2C.

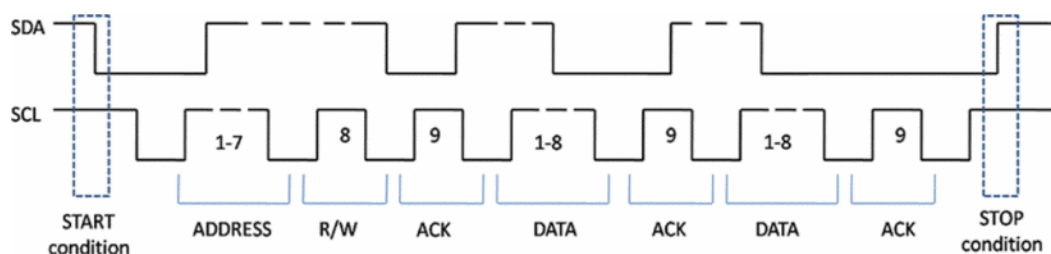


Fig. 26. Representação estrutural da comunicação I2C (Bagdalkar & Ali, 2020).

Normalmente I2C permite a comunicação entre 128 (2^n , sendo n o número de bits de endereço) dispositivos e os endereços (0 a 127) nunca deve ser duplicado e funciona a uma frequência normal de 100 kbit/s.

A comunicação é inicializada e terminada pelo dispositivo principal (master). A comunicação é inicializada quando houver uma transição de alto para baixo e termina com uma transição de baixo para alto na linha SDA enquanto a linha SCL estiver alto. Após o início da comunicação, o master envia 7 bits de endereço na linha SCL para selecionar o dispositivo secundário (slave) ao qual se pretende estabelecer a comunicação e um bit de leitura/escrita (leitura $R/\bar{W} = 1$ ou escrita $R/\bar{W} = 0$).

O barramento I2C suporta transferência de dados de apenas um byte na linha de dados SDA. Este byte pode ser um endereço de dispositivo, endereço de registo, dados gravados ou lidos do dispositivo escravo, cada bit de dados é transferido durante cada pulso de relógio de SCL começando por bit mais significativo (MSB – *Most Significant Bit*). Após a transmissão bem-sucedida de cada byte (endereço + leitura/escrita ou dados), o recetor confirma o transmissor com um sinal de confirmação, o bit ACK em que, o transmissor liberta a linha SDA para recetor

colocar a nível logico baixo (ACK). Se a linha SDA não for a nível logico baixo (NACK), a comunicação é abortada pelo master. Por defeito, a linha de SDA está ligada a uma resistência de *pull-up* para elevar a linha ao nível logico alto na ausência de sinal de dispositivos.

3.1.2. Software

Nesta secção apresentam-se alguns softwares como SolidWorks, o qual permitiu projetar, desenhar e visualizar as peças 3D para a construção da estrutura do braço robótico, o ambiente de desenvolvimento integrado (IDE – *Integrated Development Environment*) do Arduíno na programação e compilação do código para o microcontrolador ESP32 e o software MatLab para desenvolver o guia do utilizador para interface entre utilizador e o braço robótico.

3.1.2.1. Software de desenho assistido por computador - SolidWorks

O SOLIDWORKS é um software de desenho eletromecânico que permite aos utilizadores desenvolverem ideias rapidamente, produzindo modelos e desenhos detalhados. Este software 3D é bastante abrangente e eficiente, ajudando a otimizar o desenvolvimento e a fabricação de produtos como redução do tempo de desenvolvimento, aumentar a produtividade do projetista e melhorar a velocidade e a qualidade do projeto (Sam *et al.*, 2012).

O software permite a criação de um sólido ou de uma superfície para diversas áreas e dispõem de funções específicas para chapa metálica, construção soldada e moldes, proporcionando fluxo de trabalho contínuo e integrado: projeto, verificação, design, comunicação e gestão de dados (3D Lab, 2021).

3.1.2.2. Ambiente de Desenvolvimento Integrado Arduíno IDE

O ambiente de desenvolvimento integrado Arduíno IDE, é o software Arduíno de código aberto que facilita a escrita de código e o upload para as placas microcontroladores, sendo as funções escritas em linguagem de programação C ou C++. A IDE contém um editor de texto para escrever código, uma área de mensagem, uma barra de ferramentas com botões para funções comuns e uma série de menus. Este conecta-se ao hardware Arduíno para fazer upload de programas através da comunicação com PC, e permite a comunicação em tempo real entre o microcontrolador e PC via comunicação série no menu “Serial Monitor” (Arduino, 2021).

3.1.2.3. MATLAB - Matrix Laboratory

O software MATLAB é uma plataforma de programação e computação numérica usada para análise dados, desenvolvimento de algoritmos e criação de modelos. Trata-se de um software interativo de elevado desempenho voltado para o cálculo numérico, integrando análise numérica, cálculo com matrizes, processamento de sinais, construção de gráficos, interface para utilizador, sistemas de controlo, processamento de imagens e visão computacional e computação paralela, aprendizagem máquina, manutenção preditiva, robótica, teste e medição e comunicação sem fio. O programa dispõe de diversas extensões (*toolboxes* ou *blocksets*). Além dos módulos adicionais, o MATLAB conta com o Simulink, um ambiente de simulação baseado em diagrama de blocos e plataforma para *Model-Based Design* (MATLAB, 2021).

3.2. Métodos

Neste subcapítulo são apresentados os métodos e algoritmos utilizados no controlo e otimização da solução robótica desenvolvida no âmbito da dissertação.

3.2.1. Matriz de transformação Homogénea

A matriz de transformação homogénea é usada para representar a orientação e posição de um sistema de coordenadas dentro de um sistema de coordenadas "mundial" (Spatial Transformation Matrices, 2021). As subtransformações como matriz de rotação, translação, perspectiva e escala global são combinadas adequadamente numa única matriz de transformação homogénea tridimensional. A Fig. 27 representa a matriz de transformação homogénea e as respetivas subtransformações.

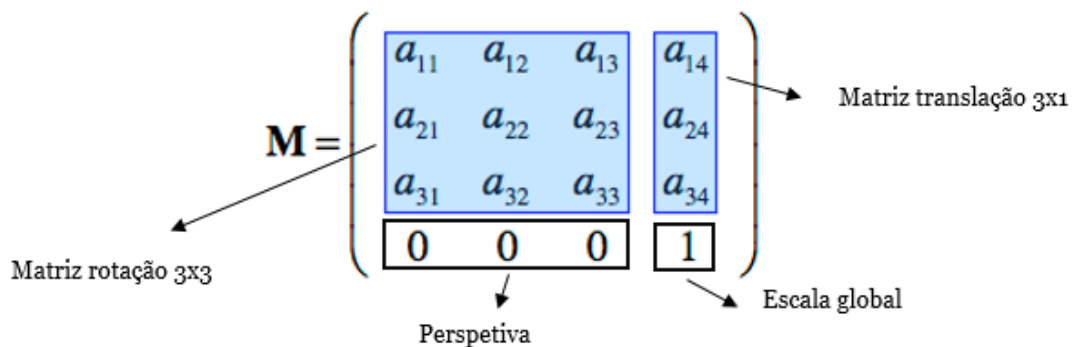


Fig. 27. Representação da matriz homogénea.

A matriz de transformação homogénea resulta de uma sequência(s) de movimento(s) de translação ou rotação segundo os eixos (x , y e z), segundo ou em torno do referencial atual (pós multiplicação) ou do referencial fixo (pré-multiplicação). A representação proporciona:

- Matriz de translação, que representa uma posição espacial (x , y e z) após deslocamento(s) segundo o(s) eixo(s) em relação ao referencial fixo e encontra-se representada nas Equações (1.a, 1.b e 1.c), para o deslocamento segundo cada um dos eixos x , y ou z :

$$T(x, d) = \begin{bmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.a)$$

$$T(y, d) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.b)$$

$$T(z, d) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.c)$$

Sendo T representa a matriz de translação, e x , y e z são os eixos de translação e d é o valor de deslocamento correspondente a translação.

- Matriz de rotação (orientação do referencial móvel em relação ao referencial fixo) representa a rotação de um ponto segundo o(s) eixo(s), a Equação (2.a, 2.b e 2.c) expressa em termos de matrizes homogêneas as matrizes de rotação para diferente eixo.

$$R(x, \alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (2.a)$$

$$R(y, \alpha) = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \quad (2.b)$$

$$R(z, \alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.c)$$

Em que α representa o ângulo em graus correspondente ao eixo de rotação x , y e z e R representa a matriz de rotação.

3.2.2. Cinemática direta

A cinemática direta permite determinar a posição e orientação da estrutura da ferramenta dados todos os seus parâmetros das juntas do manipulador e o respetivo ângulo e distância da articulação. Uma convenção comumente é usada para calcular os parâmetros exigidos no modelo cinemático em aplicações de robótica, é a convenção DH – *Denavit Hartenberg* tendo 4 parâmetros (Dasari & Reddy, 2012) associados com o elo e a junta:

- \mathbf{a}_i : comprimento do elo;
- α_i : torção do elo;
- \mathbf{d}_i : deslocamento do elo;
- θ_i : ângulo da junta;

Esta convenção dá origem a uma matriz de transformação homogénea individual como o produto de quatro transformações básicas apresentado na Equação 3.a, 3.b e 3.c (Spong *et al.*, 2006):

$$\mathbf{A}_i = \mathbf{Rot}_{Z,\theta_i} \cdot \mathbf{Trans}_{Z,d_i} \cdot \mathbf{Trans}_{X,a_i} \cdot \mathbf{Rot}_{X,\alpha_i} \quad (3. a)$$

Logo com base nas equações 1.a, 1.c, 2.a, 2.c a Equação (3.b) vem (Spong *et al.*, 2006):

$$\mathbf{A}_i = \begin{bmatrix} C_{\theta_i} & -S_{\theta_i} & 0 & 0 \\ S_{\theta_i} & C_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & d_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_{\alpha_i} & -S_{\alpha_i} & 0 \\ 0 & S_{\alpha_i} & C_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3. b)$$

Efetuada as multiplicações das matrizes, a equação (3.c) resultará em (Spong *et al.*, 2006):

$$\mathbf{A}_i = \begin{bmatrix} C_{\theta_i} & -S_{\theta_i} \cdot C_{\alpha_i} & S_{\theta_i} \cdot S_{\alpha_i} & a_i \cdot C_{\theta_i} \\ S_{\theta_i} & C_{\theta_i} \cdot C_{\alpha_i} & -C_{\theta_i} \cdot S_{\alpha_i} & a_i \cdot S_{\theta_i} \\ 0 & S_{\alpha_i} & C_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3. c)$$

Sendo \mathbf{A}_i a matriz de transformação, S seno, C cosseno e i é número de grau de liberdade.

Para i graus de liberdade a posição e orientação da estrutura da ferramenta em relação à estrutura inercial (referencial fixo) são dadas por uma matriz de transformação homogénea representado na Equação 3.d (Spong *et al.*, 2006):

$$\mathbf{A} = \begin{bmatrix} \mathbf{R}_i^0 & \mathbf{O}_i^0 \\ 0 & 1 \end{bmatrix} = \mathbf{T}_i^0(q_1, \dots, q_i) = \mathbf{A}_1(q_1) \cdot \mathbf{A}_2(q_2) \cdot \mathbf{A}_3(q_3) \cdots \mathbf{A}_i(q_i) \quad (3. d)$$

Em que A é a matriz homogênea, R matriz de rotação, q é o ângulo ou deslocamento associado a junta, O é a matriz de translação.

A transformação homogênea pode ser representada usando os 4 parâmetros DH, para tal é preciso definir e estabelecer regras para alguns referencias apresentado nas Equações 4.a, 4.b, 4.c, 4.d e 4.e (Spong et al., 2006):

- $\hat{X}_{i+1} \perp \hat{Z}_i$ (4. a)
- $\hat{X}_{i+1} \cap \hat{Z}_i$ para $i > 0$ (4. b)
- Z_i é sempre definido sobre a junta de rotação ou de translação (4. c)
- X_o é escolhido aleatoriamente (4. d)
- Y_i é escolhido segundo a regra de mão direita com X_i e Z_i (4. e)

Os valores de base física para os parâmetros DH são obtidos consoante as regras apresentado nas Equações 5.a, 5.b, 5.c e 5.d (Spong et al., 2006):

- a_i : comprimento $\{Z_i, Z_{i+1}\} X_{i+1}$ (5. a)

- α_i : ângulo $\{Z_i, Z_{i+1}\} X_{i+1}$ (5. b)

- d_i : distância $\{X_i, X_{i+1}\} Z_i$ (5. c)

- θ_i : ângulo $\{X_i, X_{i+1}\} Z_i$ (5. d)

3.2.3. Ângulo de Euler

Os ângulos de Euler são uma sequência de três ângulos (α , β e γ) que descrevem a orientação de um corpo rígido no espaço, resulta de uma rotação arbitraria em torno de três eixos (X , Y e Z) e nunca duas rotações consecutivas sobre o mesmo eixo, resultando em doze combinações validas com a sequência de três rotações elementares possíveis que são: XYZ, YZX, ZXY, XYX, YZY, ZXZ, XZY, YXZ, ZYX, XZX, YXY, ZYZ (Mecademic, 2021).

A multiplicação de três rotações elementares resulta em matriz de rotação R , por exemplo se a sequência de rotação for ZYX, a matriz de rotação é dada na Equação (6) como:

$$R = R_z(\gamma) \cdot R_y(\beta) \cdot R_x(\alpha) \quad (6)$$

Onde R é a matriz de rotação, $R_z(\gamma)$, $R_y(\beta)$ e $R_x(\alpha)$ representam rotações elementares sobre os eixos z , y e x respectivamente (nessa ordem), do referencial absoluto para ângulos (γ , β e α). Logo aplicando as equações (2.a, 2.b e 2.c) e com base na figura 26, a matriz de rotação são dadas nas Equações (7.a e 7.b) como:

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (7.a)$$

Logo, efetuando as multiplicações das matrizes, a matriz de rotação será:

$$R(\gamma, \beta, \alpha) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \cos(\gamma) \cdot \cos(\beta) & \cos(\gamma) \cdot \sin(\beta) \cdot \sin(\alpha) - \sin(\gamma) \cdot \cos(\alpha) & \cos(\gamma) \cdot \sin(\beta) \cdot \cos(\alpha) + \sin(\gamma) \cdot \sin(\alpha) \\ \sin(\gamma) \cdot \cos(\beta) & \sin(\gamma) \cdot \sin(\beta) \cdot \sin(\alpha) + \cos(\gamma) \cdot \cos(\alpha) & \sin(\gamma) \cdot \sin(\beta) \cdot \cos(\alpha) - \cos(\gamma) \cdot \sin(\alpha) \\ -\sin(\beta) & \cos(\beta) \cdot \sin(\alpha) & \cos(\beta) \cdot \cos(\alpha) \end{bmatrix} \quad (7.b)$$

3.2.4. Cinemática inversa

A cinemática inversa é um subdomínio da cinemática, calcula as posições necessárias das juntas para atingir uma posição e orientação desejada (Dasari & Reddy, 2012).

A cinemática inversa é um problema complexo e imprevisíveis relativamente à cinemática direta:

- Podem existir soluções múltiplas, ou mesmo, infinitas;
- Pode não existir solução (a posição generalizada está fora do espaço de trabalho);
- Equação não lineares, pelo que nem sempre é possível uma resolução analítica;

Em geral, a cinemática inversa requer grandes cálculos, através de métodos analíticos ou numéricos iterativos. O método analítico é aplicado quando o problema é simples e que permite a obtenção de todas as soluções, caso contrário recorre-se a métodos números iterativos (várias iterações para tentar para convergir para a solução com um determinado erro).

Aplicando a cinematiza inversa, encontrar os valores dos parâmetros da junta que colocarão a estrutura da ferramenta na posição e orientação desejadas (dentro da área de trabalho), dada uma matriz homogênea H (Equação 8.a) com as posições e orientações pretendida, é dada pela Equação 8.a (Spong et al., 2006).

$$H = \begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix} \quad (8.a)$$

para um determinado manipulador robótico com n graus de liberdade, aplicando a convenção DH, a Equação 8.b é dada por (Spong et al., 2006):

$$T_n^0(q_1, \dots, q_n) = A_1(q_1) \cdot A_2(q_2) \cdot A_3(q_3) \cdots A_n(q_n) \quad (8.b)$$

encontrar a solução (ângulo de cada junta) de modo a satisfazer a Equação 8.c é dada por (Spong et al., 2006):

$$T_n^0(q_1, \dots, q_n) = H \quad (8.c)$$

Proceder ao desacoplamento cinemático (Spong *et al.*, 2006), é ideal para sistemas que têm um braço, um pulso e com os eixos de articulação do punho alinhados em um ponto. Ou seja, para um manipulador esférico com seis graus de liberdade e que os últimos 3 eixos intersejam num ponto, é possível desacoplar o problema da cinemática inversa em dois problemas simples, conhecidos respetivamente como cinemática de posição inversa e cinemática de orientação inversa. Agora expressa-se a Equação 8.c como dois conjuntos de Equação 8.d e 8.e (Spong *et al.*, 2006) que representam as equações rotacionais e posicionais.

$$T_6^0(q_1, \dots, q_n) = R \quad (8. d)$$

$$O_6^0(q_1, \dots, q_n) = O \quad (8. e)$$

Onde O e R são a posição e orientação desejadas da ferramenta, expressas em relação ao sistema de coordenadas mundial. A Equação 8.f e 8.g (Spong *et al.*, 2006) resulta da decomposição da Equação 8.d como:

$$R = R_3^0 \cdot R_6^3 \quad (8. f)$$

$$R_6^3 = (R_3^0)^{-1} \cdot R = (R_3^0)^T \cdot R \quad (8. g)$$

Aplicando a Equação 8.g e com a matriz homogénea R conhecida, os três ângulos de junta finais podem ser encontrados como um conjunto de ângulos de Euler correspondentes a R_6^3 (Pires, 2018).

3.2.5. Velocidade e matriz Jacobiana

Quando um corpo rígido está sujeito a uma rotação pura em torno de um eixo fixo, cada ponto do corpo descreve um círculo cujo centro está sobre o eixo de rotação e cujo plano é perpendicular ao eixo. Para uma combinação de n juntas á efetuar movimentos, o efector/ferramenta irá movimentar-se com uma determinada velocidade angular (ω) e velocidade linear (v) sendo dependente da velocidade angular de cada juntas (unicamp, 2015).

O Jacobiana é a matriz formada pelas derivadas parciais de um vetor por outro vetor. Se tivermos uma função $f(x)$, o Jacobiana é a matriz das derivadas parciais para cada componente dos vetores. O Jacobiana contém todas as informações necessárias para vincular uma mudança em um componente de x a uma mudança em um componente de f . A Equação 9 representa a matriz Jacobina $J(f, x)$ generalizada (Dasari & Reddy, 2012).

$$J(\mathbf{f}, \mathbf{x}) = \frac{d\mathbf{f}}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \frac{\partial f_m}{\partial x_1} & \dots & \dots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \quad (9)$$

A matriz Jacobiana, J , representa a forma como a posição efector final é afetado pelas deslocções individuais das juntas, sendo usada em robótica para vários propósitos.

Na robótica, assumindo que se tem um manipulador com n elos com variáveis das juntas q_1, q_2, \dots, q_n , a matriz de transformação homogênea que define a posição e a orientação do efector final será dada pela Equação 10.a (Spong *et al.*, 2006):

$$T_n^0 = \begin{bmatrix} R_n^0(q) & O_n^0(q) \\ 0 & 1 \end{bmatrix} \quad (10. a)$$

a matriz Jacobiana é dada na Equação (10.b) como:

$$J = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \quad (10. b)$$

Em que J é a matriz Jacobiana com $6 \times n$, n é número de juntas, J_v é vetor Jacobiana associado a velocidade linear, J_ω associado a velocidade angular e ambos têm a dimensão matricial $3 \times n$.

Com i sendo o número da coluna de J_v e J_ω , J_{v_i} será dada na Equação (10.c) e J_{ω_i} na Equação (10.d) como (Spong *et al.*, 2006):

$$J_{v_i} = \begin{cases} Z_{i-1} \times (O_n - O_{i-1}); & \text{para } i \text{ como junta rotacional} \\ Z_{i-1} & \text{para } i \text{ como junta prismática} \end{cases} \quad (10. c)$$

$$J_{\omega_i} = \begin{cases} Z_{i-1} & \text{para } i \text{ como junta rotacional} \\ 0 & \text{para } i \text{ como junta prismática} \end{cases} \quad (10. d)$$

Sendo,

- O_j são os três primeiros elementos da última coluna de T_j^0
- Z_j é $R_j^0(k)$ ou equivalentemente, os três primeiros elementos da terceira coluna do T_j^0

Finalmente a velocidade angular ω e linear v , será calculado na equação (10.e) como:

$$v, \omega = J \cdot \dot{q} \Leftrightarrow \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} j_{v_1} & \dots & j_{v_q} \\ j_{v_1} & \dots & j_{v_q} \\ j_{v_1} & \dots & j_{v_q} \\ j_{\omega_1} & \dots & j_{\omega_q} \\ j_{\omega_1} & \dots & j_{\omega_q} \\ j_{\omega_1} & \dots & j_{\omega_q} \end{bmatrix} \cdot \begin{bmatrix} \dot{q}_1 \\ \dots \\ \dot{q}_n \end{bmatrix} \quad (10. e)$$

Sendo os vetores de velocidades (v e ω) 3×1 e os vetores Jacobiana (j_v e j_ω) $3 \times n$ e q_n são velocidade angular [m/s] de cada juntas. Uma vez tendo a matriz Jacobiana, é possível aplicar o processo inverso, ou seja, fornecer os vetores das velocidades (v e ω) e calcular velocidade angular de cada juntas ($\dot{q}_1 \dots \dot{q}_n$) conforme representado na Equação 10.f (Spong *et al.*, 2006):

$$\dot{q} = j^{-1} \cdot \vec{v} \quad (10. f)$$

Sendo \dot{q} o vetor $n \times 1$ com valores de velocidade angular de cada juntas, \vec{v} é o vetor 6×1 que contem os valores das velocidades angular e linear (v e ω). A Equação 10.f apenas é aplicável se a matriz Jacobiana admitir a inversa. Para tal esta deve ser quadrada, ou seja, ter a dimensão $6 \times n$ para $n=6$ ou seja quando o manipulado tiver 6 graus de liberdade. Quando $n \neq 6$ a matriz j não admite a inversa o que será necessário recorrer ao método da pseudo inverso (Spong *et al.*, 2006).

Para uma interpretação de cada elemento da matriz Jacobiana (Spong *et al.*, 2006), considera-se um manipulador robótico plano com duas juntas rotacionais. A Equação 10.g a matriz Jacobiana será:

$$J(f, \theta) = \begin{bmatrix} \frac{\partial f_x}{\partial \theta_1} & \frac{\partial f_x}{\partial \theta_2} \\ \frac{\partial f_y}{\partial \theta_1} & \frac{\partial f_y}{\partial \theta_2} \end{bmatrix} \quad (10. g)$$

Onde o objetivo final do efector final:

- θ representa o ângulo atual das posições das juntas.
- f representa o valor atual de posição e orientação do atuador final

Na Fig. 28 está representada a estrutura do manipulado robótico com duas rotacionais.

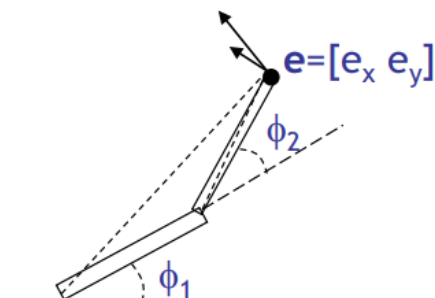


Fig. 28. Representação do manipulador plano com duas juntas rotacionais (Spong *et al.*, 2006).

3.2.6. Singularidade

A singularidade, fisicamente, num manipulador robótico ocorre quando:

- Quando existe ponto singular dentro da área do manipulador, ou seja, quando para um determinado ângulo(s) da(s) junta(s) resulta da sobreposição do elo.
- Quando a solução para um determinado ponto não pode ser encontrada (espaço de trabalho ou ângulo da junta restrito).

A singularidade depende das características do manipulador robótico e não depende do referencial escolhido para descrever a cinemática do manipulador (Pires, 2018). Para o manipulador ser capaz de atingir um determinado ponto com uma determinada velocidade arbitrária e considerando a Equação 10.e, para não atingir a singularidade matematicamente deve satisfazer a Equação 11.a (Spong *et al.*, 2006) que:

$$\text{rank}(J) \leq \min(6, n) \quad (11. a)$$

Sendo, J a matriz do Jacobiana, $rank$ o número de linha não nula resultante da combinação linear da matriz J e n é número de DOF. Dependendo da matriz, por vezes, efetuar o cálculo de rank não é simples. Para um manipulador com 6 DOF, utiliza-se a técnica de decomposição da singularidade (Pires, 2018) em singularidades do punho e do braço, em que a matriz Jacobiana 6×6 é subdividida em 4 matrizes 3×3 conforme Equação 11.c (Pires, 2018) como:

$$J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \quad (11. c)$$

As configurações singulares do punho devem ser obtidas analisando detalhadamente a estrutura do determinante da matriz J_{22} e as singularidades do braço é obtida através do determinante da matriz J_{11} . A singularidade por decomposição é verificada na Equação 11.d como:

$$\det(J) = \det(J_{11}) \cdot \det(J_{22}) = 0 \quad (11. d)$$

3.2.7. Rede Neuronal Artificial

Rede neuronal artificial é um algoritmo implementado computacionalmente, baseado no funcionamento do cérebro humano (Fernández *et al.*, 2014). É composta por neurónios ou nós, interconectados de modo que a saída de um neurónio pode ser usada como entrada de outros. A

função da rede é determinada pela ligação entre neurónios, geralmente organizadas em grupos designados por camadas (entrada, saída e oculta). A Fig. 29 representa a arquitetura de uma rede neuronal artificial, onde a camada de entrada representa o meio a qual os dados são apresentados à rede, a camada de saída mostra a resposta da rede para uma determinada entrada e a camada oculta normalmente é constituída por muitos neurónios interligados entre eles para intercalar um resultado.

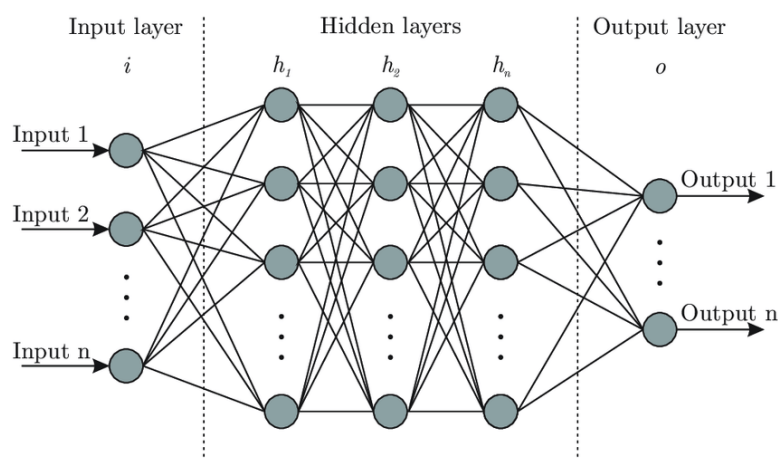


Fig. 29. Arquitetura da rede neuronal artificial (Fernández *et al.*, 2014).

A rede neuronal artificial apresenta muitas valias em diferentes áreas de engenharias, bem como auxílio na resolução e otimização dos problemas do quotidiano. É proposto utilizar uma rede neuronal artificial como meio previsionar de deteção de possível colisão através de aumento da corrente elétrica. A rede será treinada com base no tipo de movimento, posição e orientação a alcançar, velocidade e a carga útil e sempre a monitorizar o estado da corrente elétrica.

3.3. Nota conclusiva

Ao longo deste capítulo foram apresentadas alguns matérias, software e processos para a construção do manipulador robótico. Também para controlo e movimento, foram apresentados e abordados temas como microcontrolador, motor de passo e encoder. Na cinemática direta, aplicando a convenção DH permite obter rapidamente a matriz de transformação homogénea que relaciona a posição e orientação do efector sem ter que efetuar cálculos complexos ficando pendente do número de graus de liberdade do manipulador.

O ângulo de Euler quando é aplicado à cinemática direta permite transformar a matriz de rotação (retirada na matriz homogénea final, Equação 4) em orientação angular (γ , β e α) e quando é

aplicada à cinemática inversa, permite converter orientação angular (γ , β e α) em matriz de rotação.

A cinemática inversa requer cálculos morosos (exige muito tempo) e complexos e quando se aplica ao método numérico iterativo pode tornar o sistema mais oneroso ou com atrasos devido o processamento.

4. Protótipo e Dimensionamento do Braço Robótico

Neste capítulo são apresentadas as peças, processos de montagem e estruturas final do braço robótico. Também são apresentados os cálculos matemáticos da convenção DH, dimensionamento do circuito elétrico/eletrônico e bem como a interface do utilizador com o manipulador robótico final.

4.1. Peças e estrutura do braço robótico

A Fig. 30 apresenta as peças e estrutura 3D final do braço robótico projetado e alterado no software SolidWorks com base no modelo Robô WE-R2.4 (Thingiverse, 2020).

A peça designada por base cônica (Figura 30.a) é o suporte/base do manipulado, conter o motor do eixo 1 e também no seu topo é fixado o conjunto das engrenagem e suporte para fazer transmissão para eixo 2.

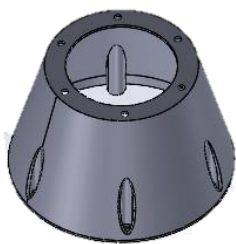
O anel dentado (Figura 30.b) é responsável para interligar dois elos, trata-se de uma peça movel afetado pelo movimento do motor e pinhão.

A peça designada por ombro (Figura 30.c) trata de suporte para motor para eixo 2 e em diante, é fixo no anel dentado e também serve de suporte para anel dentado fixo. O braço de elevação (Figura 30.d) permite a extensão do elo do eixo 2 e 3 de modo que o efector final tenha maior alcance.

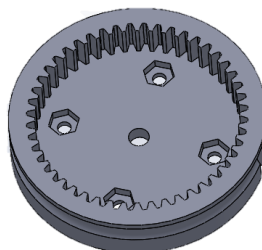
O anel dentado com fixação para motor (Figura 30.e) permite fixar o motor, proporcionar o contacto mecânico entre o pinhão acoplado no veio do motor e o pinhão duplo movel que por sua vez proporcional o contacto com o anel dentado. O pinhão duplo secundário (Figura 30.e) garante a transmissão mecânica do pinhão do motor para o anel dentado.

A peça ombreira (Figura 30.g) serve para garantir de suporte e ligação entre anel dentado e o Ombo. O pinhão simples (Figura 30.h) está acoplado no eixo de cada motor estabelecendo o contacto mecânico rotativo entre motor e o pinhão duplo secundário. A roda dentada (Figura 30.i), fixa no pinhão simples principal serve de apoio e estabilizador entre o pinhão simples principal e os pinhões duplos.

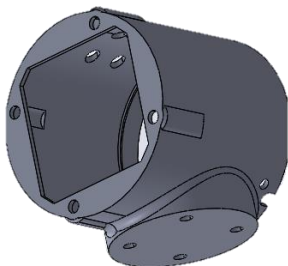
O braço robótico (Figura 30.j) trata-se do protótipo original sem antes sofrer alteração/adaptação de acordo ao projeto implementado.



a) Base cónica do manipulador



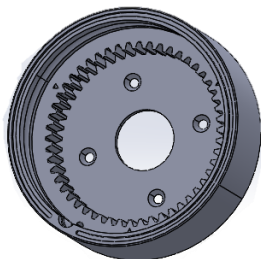
b) Anel dentado



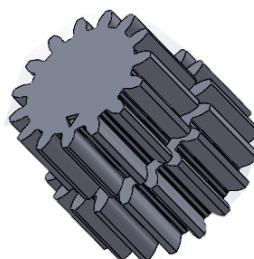
c) Ombro



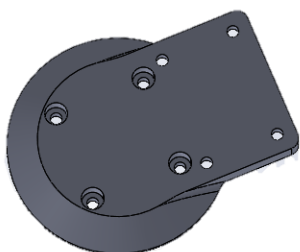
d) Braço elevação



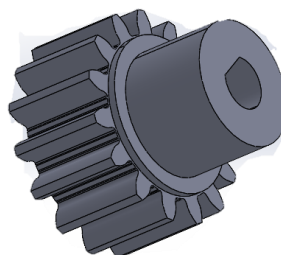
e) Anel dentado com fixação para motor



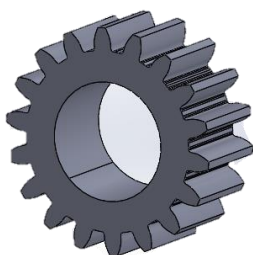
f) Pinhão duplo secundário



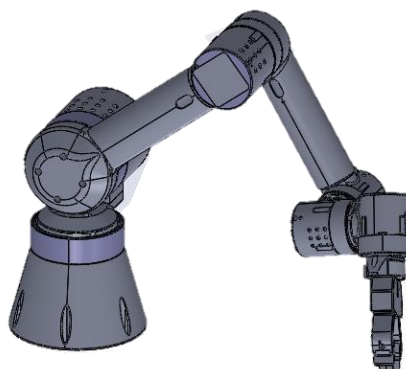
g) Ombreira



h) Pinhão simples principal



i) Roda dentada



j) Braço robótico

Fig. 30. Peças e estrutura 3D do braço robótico.

4.2. Formulação matemática da convenção – DH e cinemática direta/inversa

Neste subcapítulo é determinada a cinemática do manipulador, estabelecendo as regras para referências aplicando as Equações 4. A Fig. 31 apresenta a cinemática do manipulador e na Tabela 10. os valores das distâncias entre elos.

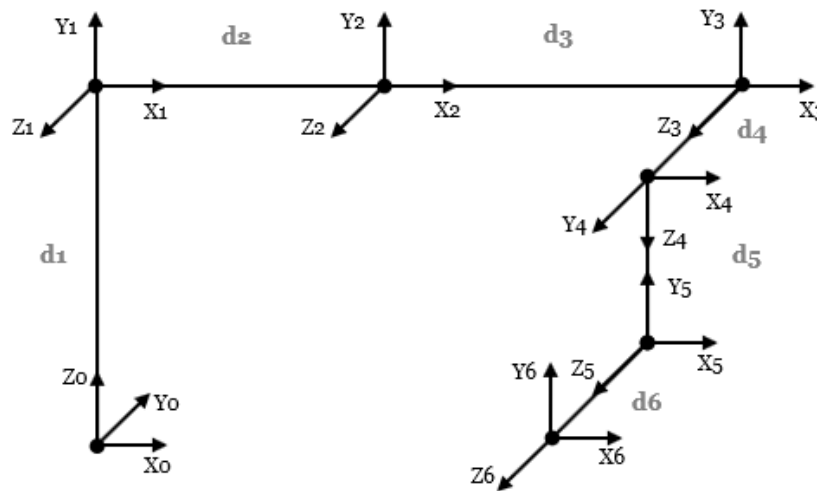


Fig. 31. Cinemática do manipulador robótico.

Tabela 11. Valores das distâncias entre elos.

d	Distância [mm]
d ₁	163,00
d ₂	204,00
d ₃	230,00
d ₄	100,00
d ₅	100,00
d ₆	100,00

Posteriormente obtém-se os parâmetros DH apresentado na Tabela 11. aplicando as Equações 5.

Tabela 12. Parâmetros DH do manipulador robótico.

Elo	a [mm]	α [°]	d [mm]	θ [°]
L ₀₁	0	90	d ₁	0 + θ_1
L ₁₂	d ₂	0	0	0 + θ_2
L ₂₃	d ₃	0	0	0 + θ_3
L ₃₄	0	90	d ₄	0 + θ_4
L ₄₅	0	-90	d ₅	0 + θ_5
L ₅₆	0	0	d ₆	0 + θ_6

Aplicando a Equação 3.c e com os parâmetros DH da Tabela 11, obtêm-se as matrizes de transformação ($A_1, A_2, A_3, A_4, A_5, A_6$) e denotado C como cosseno e S seno obtemos:

$$A_1 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & 0 \\ S\theta_1 & 0 & -C\theta_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. a) \quad A_2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & d_2 \cdot C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & d_2 \cdot S\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. b)$$

$$A_3 = \begin{bmatrix} C\theta_3 & -S\theta_3 & 0 & d_3 \cdot C\theta_3 \\ S\theta_3 & C\theta_3 & 0 & d_3 \cdot S\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. c) \quad A_4 = \begin{bmatrix} C\theta_4 & 0 & S\theta_4 & 0 \\ S\theta_4 & 0 & -C\theta_4 & 0 \\ 0 & 1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. d)$$

$$A_5 = \begin{bmatrix} C\theta_5 & 0 & -S\theta_5 & 0 \\ S\theta_5 & 0 & C\theta_5 & 0 \\ 0 & -1 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. e) \quad A_6 = \begin{bmatrix} C\theta_6 & -S\theta_6 & 0 & 0 \\ S\theta_6 & C\theta_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12. f)$$

As matrizes homogêneas parciais e final (A_n^0) será dada como produto das matrizes de transformação correspondente na Equação 3.d como:

$$A_1^0 = A_1 = \begin{bmatrix} C\theta_1 & 0 & S\theta_1 & 0 \\ S\theta_1 & 0 & -C\theta_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. a)$$

$$A_2^0 = A_1^0 \cdot A_2 = \begin{bmatrix} C\theta_1 \cdot C\theta_2 & -C\theta_1 \cdot S\theta_2 & S\theta_1 & d_2 \cdot C\theta_1 \cdot C\theta_2 \\ C\theta_2 \cdot S\theta_1 & -S\theta_1 \cdot S\theta_2 & -C\theta_1 & d_2 \cdot C\theta_2 \cdot S\theta_1 \\ S\theta_2 & C\theta_2 & 0 & d_1 + d_2 \cdot S\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. b)$$

$$A_3^0 = \begin{bmatrix} C(\theta_2 + \theta_3) \cdot C\theta_1 & -S(\theta_2 + \theta_3) \cdot C\theta_1 & S\theta_1 & C\theta_1 \cdot d_3 \cdot C(\theta_2 + \theta_3) + d_2 \cdot C\theta_2 \\ C(\theta_2 + \theta_3) \cdot S\theta_1 & -S(\theta_2 + \theta_3) \cdot S\theta_1 & -C\theta_1 & S\theta_1 \cdot d_3 \cdot C(\theta_2 + \theta_3) + d_2 \cdot C\theta_2 \\ S(\theta_2 + \theta_3) & C(\theta_2 + \theta_3) & 0 & d_1 + d_3 \cdot S(\theta_2 + \theta_3) + d_2 \cdot S\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. c)$$

$$A_4^0 = \begin{bmatrix} C(\theta_2 + \theta_3 + \theta_4) \cdot C\theta_1 & S\theta_1 & S(\theta_2 + \theta_3 + \theta_4) \cdot C\theta_1 & A_4^0(1,4) \\ C(\theta_2 + \theta_3 + \theta_4) \cdot S\theta_1 & -C\theta_1 & S(\theta_2 + \theta_3 + \theta_4) \cdot S\theta_1 & A_4^0(2,4) \\ S(\theta_2 + \theta_3 + \theta_4) & 0 & -C(\theta_2 + \theta_3 + \theta_4) & A_4^0(3,4) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. d)$$

$$A_4^0(1,4) = d_4 \cdot S\theta_1 + d_2 \cdot C\theta_1 \cdot C\theta_2 + d_3 \cdot C\theta_1 \cdot C\theta_2 \cdot C\theta_3 - d_3 \cdot C\theta_1 \cdot S\theta_2 \cdot S\theta_3 \quad (13. d. 1)$$

$$A_4^0(2,4) = d_2 \cdot C\theta_2 \cdot S\theta_1 - d_4 \cdot C\theta_1 + d_3 \cdot S\theta_1 \cdot C\theta_2 \cdot C\theta_3 - d_3 \cdot S\theta_1 \cdot S\theta_2 \cdot S\theta_3 \quad (13. d. 2)$$

$$A_4^0(3,4) = d_1 + d_3 \cdot S(\theta_2 + \theta_3) + d_2 \cdot S\theta_2 \quad (13. d. 3)$$

$$A_5^0 = \begin{bmatrix} S\theta_1 \cdot S\theta_5 + C\theta_1 \cdot C\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) & -C\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) & A_5^0(1,3) & A_4^0(1,4) \\ S\theta_1 \cdot C\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) - C\theta_1 \cdot S\theta_5 & -S\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) & A_5^0(2,3) & A_4^0(2,4) \\ C\theta_5 \cdot S(\theta_2 + \theta_3 + \theta_4) & C(\theta_2 + \theta_3 + \theta_4) & A_5^0(3,3) & A_4^0(3,4) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. e)$$

$$A_5^0(1,3) = C\theta_5 \cdot S\theta_1 - C\theta_1 \cdot S\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) \quad (13. e. 1)$$

$$A_5^0(2,3) = -C\theta_1 \cdot C\theta_5 - S\theta_1 \cdot S\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) \quad (13. e. 2)$$

$$A_5^0(3,3) = -S\theta_5 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. e. 3)$$

$$A_5^0(1,4) = d_4 \cdot S\theta_1 + d_5 \cdot C\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) + d_2 \cdot C\theta_1 \cdot C\theta_2 + d_3 \cdot C\theta_1 \cdot C\theta_2 \cdot C\theta_2 - d_3 \cdot C\theta_1 \cdot S\theta_2 \cdot S\theta_3 \quad (13. e. 4)$$

$$A_5^0(2,4) = d_5 \cdot S\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) - d_4 \cdot C\theta_1 + d_2 \cdot C\theta_2 \cdot S\theta_1 + d_3 \cdot C\theta_2 \cdot C\theta_3 \cdot S\theta_1 - d_3 \cdot S\theta_1 \cdot S\theta_2 \cdot S\theta_3 \quad (13. e. 5)$$

$$A_5^0(3,4) = d_1 + d_3 \cdot S(\theta_2 + \theta_3) + d_2 \cdot S\theta_2 - d_5 \cdot C(\theta_2 + \theta_3 + \theta_4) \quad (13. e. 6)$$

$$A_6^0 = \begin{bmatrix} A_6^0(1,1) & A_6^0(1,2) & A_6^0(1,3) & A_6^0(1,4) \\ A_6^0(2,1) & A_6^0(2,2) & A_6^0(2,3) & A_6^0(2,4) \\ A_6^0(3,1) & A_6^0(3,2) & A_6^0(3,3) & A_6^0(3,4) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13. f)$$

$$A_6^0(1,1) = C\theta_6 \cdot S\theta_1 \cdot S\theta_5 + C\theta_1 \cdot C\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) - C\theta_1 \cdot S\theta_6 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 1)$$

$$A_6^0(2,1) = -C\theta_6 \cdot C\theta_1 \cdot S\theta_5 - C\theta_5 \cdot S\theta_1 \cdot C(\theta_2 + \theta_3 + \theta_4) - S\theta_1 \cdot S\theta_6 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 2)$$

$$A_6^0(3,1) = S\theta_6 \cdot C(\theta_2 + \theta_3 + \theta_4) + C\theta_5 \cdot C\theta_6 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 3)$$

$$A_6^0(1,2) = -S\theta_6 \cdot S\theta_1 \cdot S\theta_5 + C\theta_1 \cdot C\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) - C\theta_1 \cdot C\theta_6 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 4)$$

$$A_6^0(2,2) = S\theta_6 \cdot C\theta_1 \cdot S\theta_5 - C\theta_5 \cdot S\theta_1 \cdot C(\theta_2 + \theta_3 + \theta_4) - C\theta_6 \cdot S\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 5)$$

$$A_6^0(3,2) = C\theta_6 \cdot C(\theta_2 + \theta_3 + \theta_4) - C\theta_5 \cdot S\theta_6 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 6)$$

$$A_6^0(1,3) = C\theta_5 \cdot S\theta_1 - C\theta_1 \cdot S\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 7)$$

$$A_6^0(2,3) = -C\theta_1 \cdot C\theta_5 - S\theta_1 \cdot S\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 8)$$

$$A_6^0(3,3) = -S\theta_5 \cdot S(\theta_2 + \theta_3 + \theta_4) \quad (13. f. 9)$$

$$A_6^0(1,4) = d_4 \cdot S\theta_1 + d_6 \cdot C\theta_5 \cdot S\theta_1 - C\theta_1 \cdot S\theta_5 \cdot C(\theta_2 + \theta_3 + \theta_4) + d_5 \cdot C\theta_1 \cdot S(\theta_2 + \theta_3 + \theta_4) + d_2 \cdot C\theta_1 \cdot C\theta_2 + d_3 \cdot C\theta_1 \cdot C\theta_2 \cdot C\theta_3 - d_3 \cdot C\theta_1 \cdot S\theta_2 \cdot S\theta_3 \quad (13. f. 10)$$

$$A_6^0(2,4) = d5 \cdot S\theta1 \cdot S(\theta2 + \theta3 + \theta4) - d4 \cdot C\theta1 + d2 \cdot C\theta2 \cdot S\theta1 + d3 \cdot C\theta2 \cdot C\theta3 \cdot S\theta1 - d3 \cdot S\theta1 \cdot S\theta2 \cdot S\theta3 \quad (13. f. 11)$$

$$A_6^0(3,4) = d1 + d3 \cdot S(\theta2 + \theta3) + d2 \cdot S\theta2 - d5 \cdot C(\theta2 + \theta3 + \theta4) \quad (13. f. 12)$$

Matriz Jacobiana

Para determinar a matriz do Jacobiana relembrem-se as Equação 10.c e Equação 10.d. A expressão da matriz jacobiana aplicada ao manipulador é dada na Equação 14.a como:

$$J(q) = \begin{bmatrix} z_0x(O_6 - O_0) & z_1x(O_6 - O_1) & z_2x(O_6 - O_2) & z_3x(O_6 - O_3) & z_4x(O_6 - O_4) & z_5x(O_6 - O_5) \\ Z_0 & Z_1 & Z_2 & Z_3 & Z_4 & Z_5 \end{bmatrix} \quad (14. a)$$

A matriz O_i representada na Equação 14.b corresponde aos três primeiros elementos da última coluna da matriz A_i^0 apresentados na Equação 13.

$$O_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (14. b. 1)$$

$$O_1 = \begin{bmatrix} 0 \\ 0 \\ d1 \end{bmatrix} \quad (14. b. 2)$$

$$O_2 = \begin{bmatrix} d2 \cdot C\theta1 \cdot C\theta2 \\ d2 \cdot C\theta2 \cdot S\theta1 \\ d1 + d2 \cdot S\theta2 \end{bmatrix} \quad (14. b. 3)$$

$$O_3 = \begin{bmatrix} C\theta1 \cdot d3 \cdot C(\theta2 + \theta3) + d2 \cdot C\theta2 \\ S\theta1 \cdot d3 \cdot C(\theta2 + \theta3) + d2 \cdot C\theta2 \\ d1 + d3 \cdot S(\theta2 + \theta3) + d2 \cdot S\theta2 \end{bmatrix} \quad (14. b. 4)$$

$$O_4 = \begin{bmatrix} d4 \cdot S\theta1 + d2 \cdot C\theta1 \cdot C\theta2 + d3 \cdot C\theta1 \cdot C\theta2 \cdot C\theta3 - d3 \cdot C\theta1 \cdot S\theta2 \cdot S\theta3 \\ d2 \cdot C\theta2 \cdot S\theta1 - d4 \cdot C\theta1 + d3 \cdot S\theta1 \cdot C\theta2 \cdot C\theta3 - d3 \cdot S\theta1 \cdot S\theta2 \cdot S\theta3 \\ d1 + d3 \cdot S(\theta2 + \theta3) + d2 \cdot S\theta2 \end{bmatrix} \quad (14. b. 5)$$

$$O_5 = \begin{bmatrix} d4 \cdot S\theta1 + d5 \cdot C\theta1 \cdot S(\theta2 + \theta3 + \theta4) + d2 \cdot C\theta1 \cdot C\theta2 + d3 \cdot C\theta1 \cdot C\theta2 \cdot C\theta3 - d3 \cdot C\theta1 \cdot S\theta2 \cdot S\theta3 \\ d5 \cdot S\theta1 \cdot S(\theta2 + \theta3 + \theta4) - d4 \cdot C\theta1 + d2 \cdot C\theta2 \cdot S\theta1 + d3 \cdot C\theta2 \cdot C\theta3 \cdot S\theta1 - d3 \cdot S\theta1 \cdot S\theta2 \cdot S\theta3 \\ d1 + d3 \cdot S(\theta2 + \theta3) + d2 \cdot S\theta2 - d5 \cdot C(\theta2 + \theta3 + \theta4) \end{bmatrix} \quad (14. b. 6)$$

$$O_6 = \begin{bmatrix} O_6(1,1) \\ d5 \cdot S\theta1 \cdot S(\theta2 + \theta3 + \theta4) - d4 \cdot C\theta1 + d2 \cdot C\theta2 \cdot S\theta1 + d3 \cdot C\theta2 \cdot C\theta3 \cdot S\theta1 - d3 \cdot S\theta1 \cdot S\theta2 \cdot S\theta3 \\ d1 + d3 \cdot S(\theta2 + \theta3) + d2 \cdot S\theta2 - d5 \cdot C(\theta2 + \theta3 + \theta4) \end{bmatrix} \quad (14. b. 7)$$

$$O_6(1,1) = d4 \cdot S\theta1 + d6 \cdot C\theta5 \cdot S\theta1 - C\theta1 \cdot S\theta5 \cdot C(\theta2 + \theta3 + \theta4) + d5 \cdot C\theta1 \cdot S(\theta2 + \theta3 + \theta4) + d2 \cdot C\theta1 \cdot C\theta2 + d3 \cdot C\theta1 \cdot C\theta2 \cdot C\theta3 - d3 \cdot C\theta1 \cdot S\theta2 \cdot S\theta3 \quad (14. b. 7.1)$$

A matriz Z_i representada na Equação 14.c corresponde aos três primeiros elementos da terceira coluna da matriz A_i^0 apresentada na Equação 13.

$$Z_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (14. c. 1)$$

$$Z_1 = \begin{bmatrix} S\theta1 \\ -C\theta1 \\ 0 \end{bmatrix} \quad (14. c. 2)$$

$$Z_2 = \begin{bmatrix} S\theta1 \\ -C\theta1 \\ 0 \end{bmatrix} \quad (14. c. 3)$$

$$Z_3 = \begin{bmatrix} S\theta1 \\ -C\theta1 \\ 0 \end{bmatrix} \quad (14. c. 4)$$

$$Z_4 = \begin{bmatrix} S(\theta2 + \theta3 + \theta4) \cdot C\theta1 \\ S(\theta2 + \theta3 + \theta4) \cdot S\theta1 \\ -C(\theta2 + \theta3 + \theta4) \end{bmatrix} \quad (14. c. 5)$$

$$Z_5 = \begin{bmatrix} C\theta5 \cdot S\theta1 - C\theta1 \cdot S\theta5 \cdot C(\theta2 + \theta3 + \theta4) \\ -C\theta1 \cdot C\theta5 - S\theta1 \cdot S\theta5 \cdot C(\theta2 + \theta3 + \theta4) \\ -S\theta5 \cdot S(\theta2 + \theta3 + \theta4) \end{bmatrix} \quad (14. c. 6)$$

Cinemática inversa

A cinemática inversa é um problema importante e desafiador no campo da robótica. A solução analítica do problema é complexa e demorada dependendo da configuração do manipulador e de números de soluções. Aplicou-se o método heurístico iterativos resolver o problema da cinemática inversa recorrendo ao algoritmo de otimização enxame de partículas (*Particle Swarm Optimization* – PSO, sendo utilizado por (Alkayyali & Tutunji, 2019) obtendo bom resultado em relação a outra meta heurística existe na literatura.

O algoritmo proposto utiliza PSO para gerar soluções para cinemáticas inversas através da busca no espaço, ou seja, encontra os valores ideais dos 6 ângulos das articulações para minimizar o erro. A Figura 32 representa pseudocódigo do algoritmo PSO utilizado no processo da cinemática inversa.

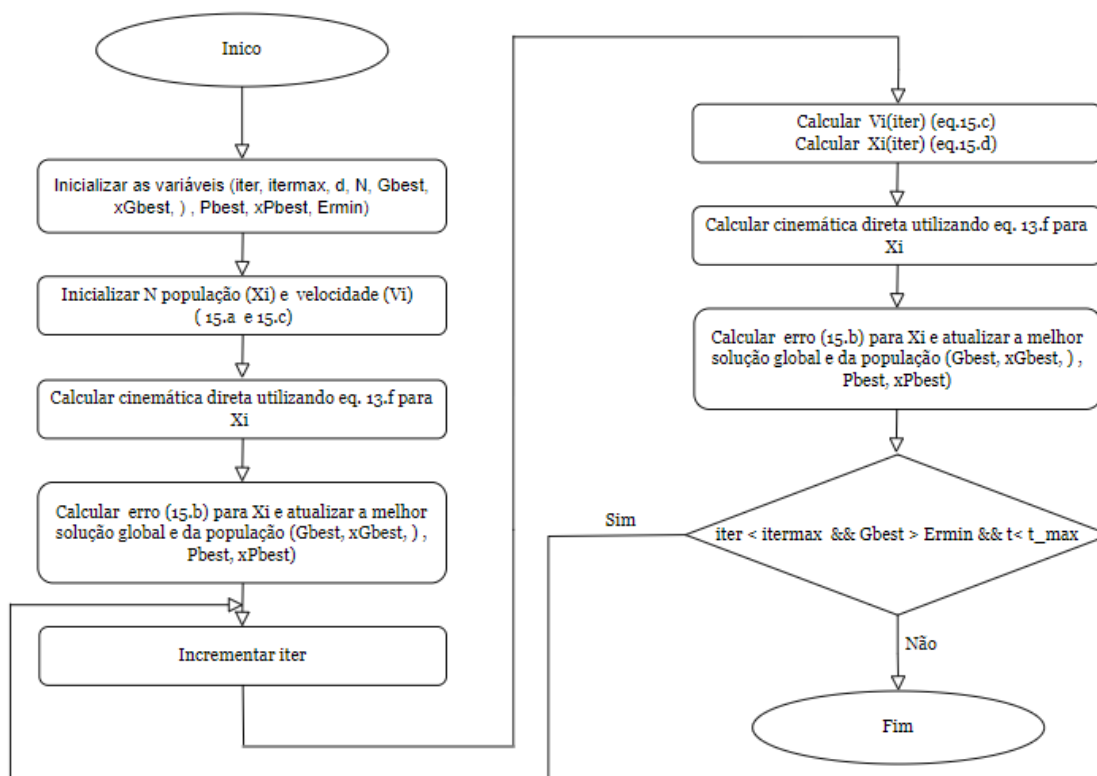


Fig. 32. Pseudocódigo para meta heurística PSO.

A Equação 15.a corresponde ao cálculo aleatório da posição inicial das partículas dentro de um determinado limite (L_{sup} e L_{inf}), que nesta aplicação resulta no cálculo do ângulo inicial para cada junta dentro de seu limite de alcance predefinido. A Equação 15.b (erro) aplicado por Alkayali & Tutunji (2019) corresponde ao erro entre a posição e orientação calculada e a posição e orientação pretendida. Sendo a posição e a orientação calculadas associadas diretamente com o ângulo de cada junta calculado pelo algoritmo. A Equação 15.c, designada por velocidade a que partícula irá movimentar-se, ou seja, dependendo da diversificação e de intensificação.

$$X_i = L_{inf} + (L_{sup} - L_{inf}) \cdot r \quad (15. a)$$

$$Erro = \rho \cdot E_p + (1 - \rho)E_r \quad (15. b)$$

$$V_{i,d}^{iter+1} = K[W \cdot V_{i,d}^{iter} + c_1 \cdot r_1 \cdot (xP_{best,i,d} - X_{i,d}^{iter}) + c_2 \cdot r_2 \cdot (x_{best} - X_{i,d}^{iter})] \quad (15.c)$$

$$X_{i,d}^{iter+1} = X_{i,d}^{iter} + V_{i,d}^{iter+1} \quad (15.d)$$

$$E_p = \frac{\|P_{PSO} - P_d\|^2}{\|P_d\|^2} \quad (15.e)$$

$$E_r = \frac{\|R_{PSO} - R_d\|^2}{\|R_d\|^2} \quad (15.f)$$

$$\rho = 0.7 \cdot e^{-E_r} + 0.3 \quad (15.g)$$

$$K = \frac{2}{|2 - \rho - \sqrt{\rho^2 - 4 \cdot \rho}|} \quad (15.i)$$

$$\varphi = c_1 + c_2, \varphi > 4, \text{tipicamente } \varphi = 4.1 \text{ e } K = 0.7298 \quad (15.j)$$

$$W = W_{max} - \frac{W_{max} - W_{min}}{itermax} \cdot iter \quad (15.k)$$

Sendo:

- i - número de partículas;
- d - dimensão do problema (6 eixos);
- E_{min} - erro mínimo. Este parâmetro é configurável pelo utilizado e quanto menor for o erro, maior será o tempo de processamento na procura da melhor solução e mais próximo será a posição alcançada em relação à posição pretendida. De acordo com Alkayyali & Tutunji (2019), este parâmetro está definido para 0,1 % do erro inicial.
- t / t_{max} - tempo simulação/tempo máximo de simulação;
- $iter / itermax$ - iteração atual / máxima;
- xG_{best} / xP_{best} - melhor solução global / melhor solução individual pela partículas;
- X_i - posição atual da partícula i ;
- V_i - velocidade da partícula i ;
- L_{inf} / L_{sup} - limites inferiores / superiores do espaço de procura;
- Erro - erro em relação à posição e orientação do final do efector;
- E_p / E_r - erro associado a posição / orientação;

- P_{PSO} / R_{PSO} - posição/orientação associada a partículas; (Calculado por PSO)
- P_d / R_d - posição/orientação pretendido;
- W - fator de inércia entre [0.4 0.9], sendo, $W_{min}=0.4$ e $W_{max}=0.9$;
- r_1, r_2 - números aleatórios entre [0 1];
- c_1, c_2 - fatores sociais e cognitivos aleatórios entre [2 4];
- K - coeficiente de restrição;
- φ e ρ - constantes.

De acordo com Alkayali & Tutunji (2019), o número de elementos da população (N) pertence ao intervalo [1 , 1000] e iteração máxima de 1000. Nesta aplicação é definido $N = 150$ (após vários testes, a convergência era garantida com $N < 150$), e o número máximo de iterações (itermax) igual 1000. De modo que o tempo de convergência seja reduzido, define-se o tempo máximo de processamento para 10 [s]. Estes parâmetros podem ser alterados no interface do utilizador.

4.3. Circuito elétrico, controlo e treino da rede neuronal

Neste capítulo apresenta-se o dimensionamento do circuito elétrico de controlo e a estrutura de comunicação.

Na robótica assiste-se a uma grande dificuldade em reduzir a cablagem no braço robótico, de modo a proporcionar-lhe maior mobilidade. Por outro lado, o processamento das informações requer um tempo reduzido de modo a proporcionar uma rápida resposta.

Pelas especificações do microcontrolador ESP, este apresenta características robustas a nível de processamento, fiabilidade e protocolo de comunicação, pelo que são utilizados:

- Um ESP32 principal, que estabelece a comunicação entre PC (Interface do utilizador - MatLab) via cabo USB. O ESP32 principal também estabelece a comunicação sem fios utilizando o protocolo ESP NOW entre os 6 ESP's de cada eixo do braço robótico.
- Os 6 ESP32s de cada eixo estabelecem a comunicação sem fios bidirecional com o ESP32 principal, comunicando cada um à vez. Esta técnica de comunicação sem fios auxilia na diminuição da cablagem entre ESP32s e o ESP32 principal. De um modo geral, diminui o peso da cablagem e facilita a mobilidade do braço. Para tal, apenas existe cablagem elétrica para alimentação elétrica de cada eixo.

A Fig. 33 apresenta a estrutura de comunicação entre os ESP32 (principal e secundários) e PC.

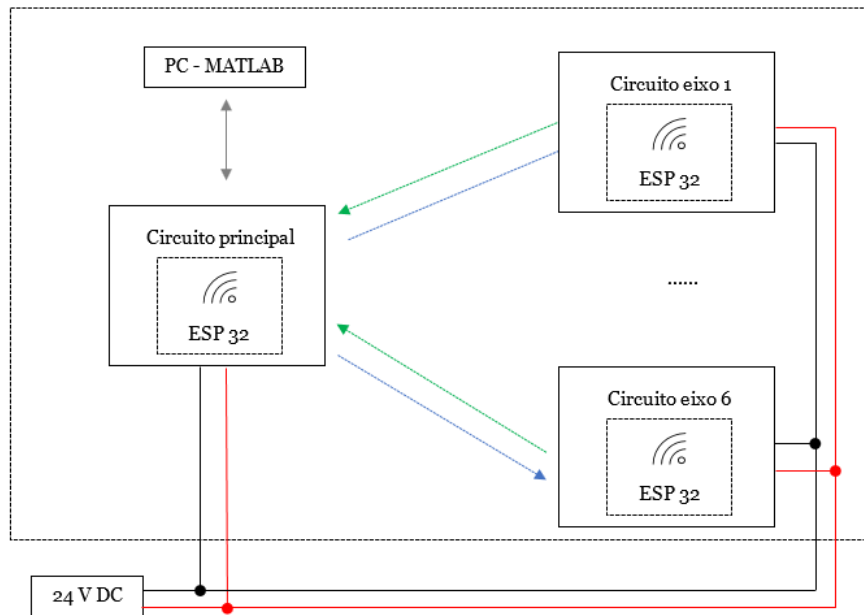


Fig. 33. Esquema e tipologia de comunicação entre os ESP32 e PC.

Na Fig. 34 está representada a estrutura de comunicação entre o ESP32 principal e PC (interface do utilizador) de modo a enviar e receber dados, movimentar o braço robótico e visualizar o estado do braço. Esta comunicação via cabo é garantida utilizando um conversor UART para USB.

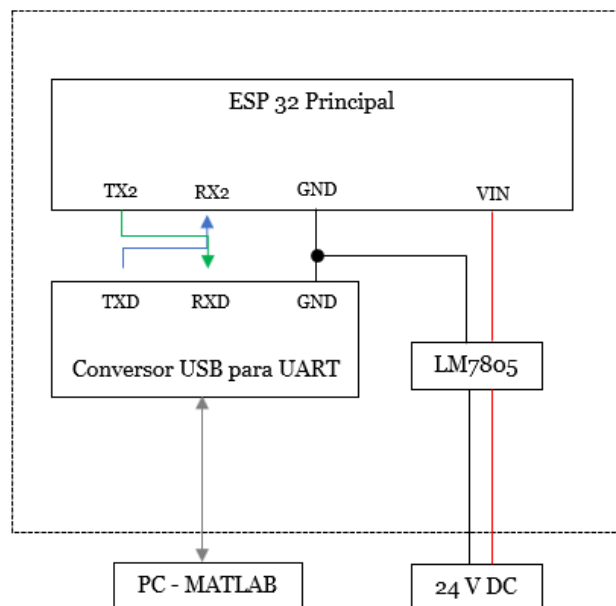


Fig. 34. Esquema de comunicação entre o ESP32 principal e PC.

A Fig. 35 representa a ligação elétrica entre ESP32, encoder, conversor analógico digital (ADS1115), amplificador operacional e a drive do motor de passo (TIC 500). As ligações são estabelecidas através do terminal designado por “J1, J2 e J3”. O terminal J1 com 6 terminais recebe a alimentação elétrica para PCB do respetivo eixo, ao drive do motor (TIC 500) e dispõe também da alimentação elétrica para o eixo seguinte. O terminal J2 de 4 terminais estabelece a ligação elétrica (5V para ESP32 e encoder) e a comunicação UART entre o ESP32 do eixo atual e o circuito do drive do motor (TIC 500). Por fim, o terminal J3 também de 4 terminais, estabelece a ligação elétrica (5V) e os sinais (A+ e B+) do encoder e o ESP32. Esta técnica de usar os terminais de ligação na PCB resulta num rápido isolamento entra as partes.

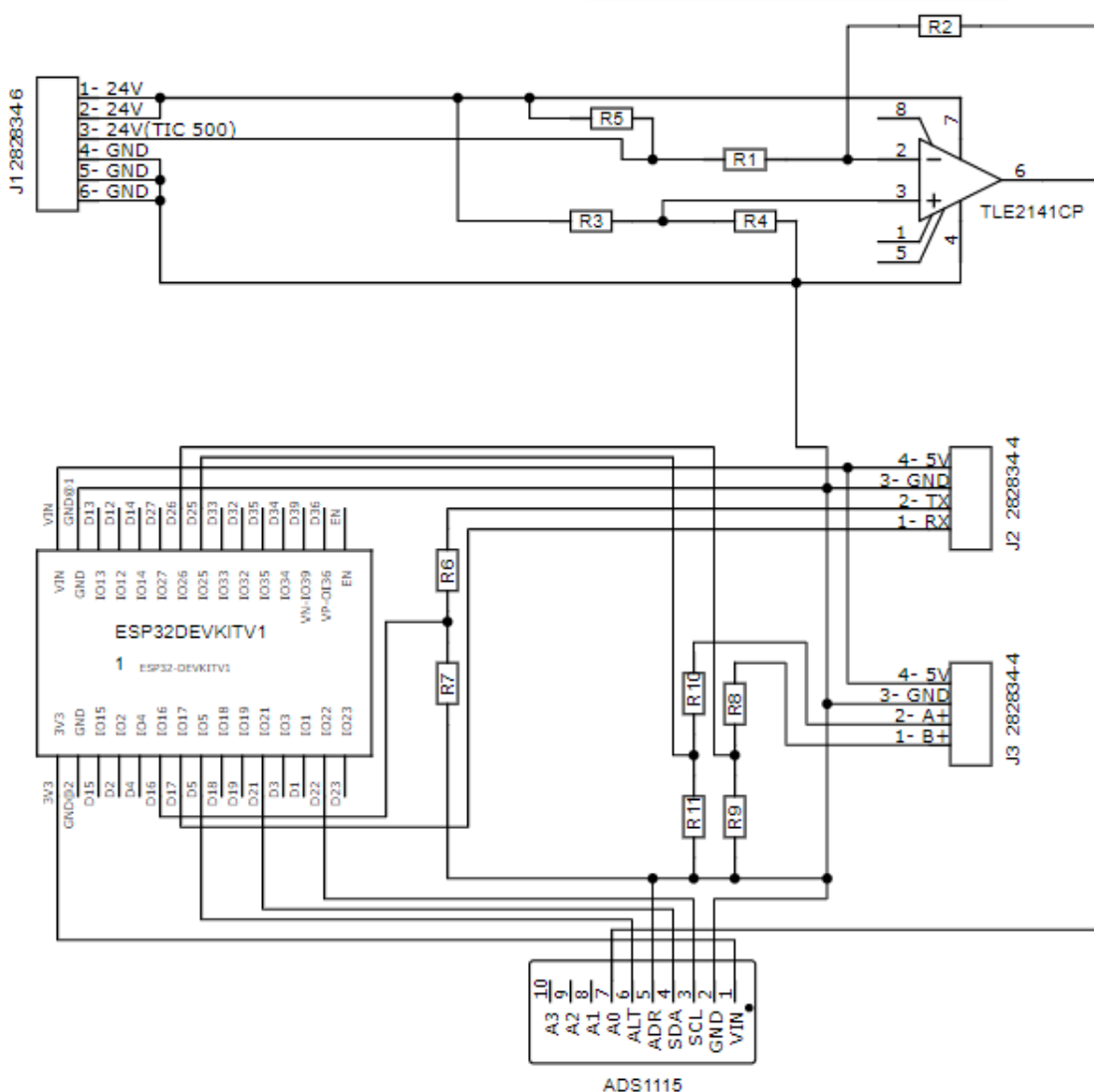
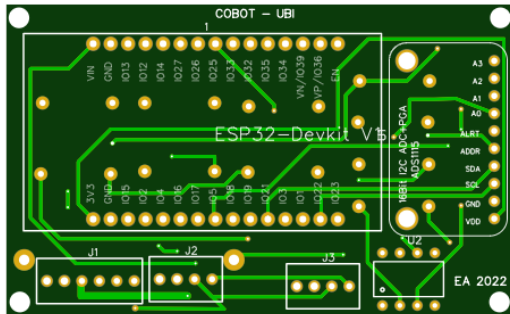
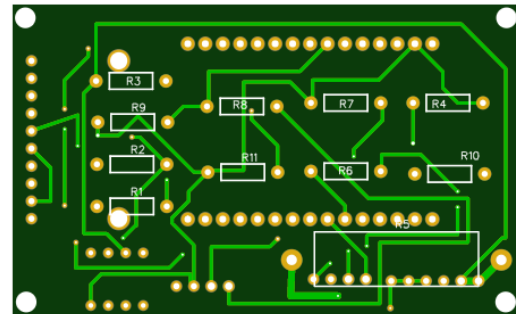


Fig. 35. Esquema de ligação entre os dispositivos de cada eixo.

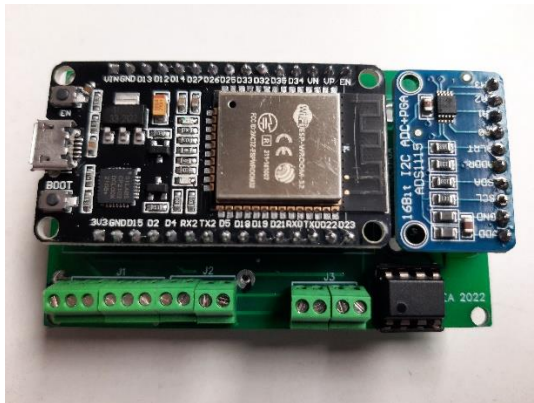
De forma a interligar os componentes do circuito da Fig. 35 e minimizar a área, foi desenvolvida e impressa uma placa circuito impresso (PCB - *printed circuit board*) representada na Fig. 36.



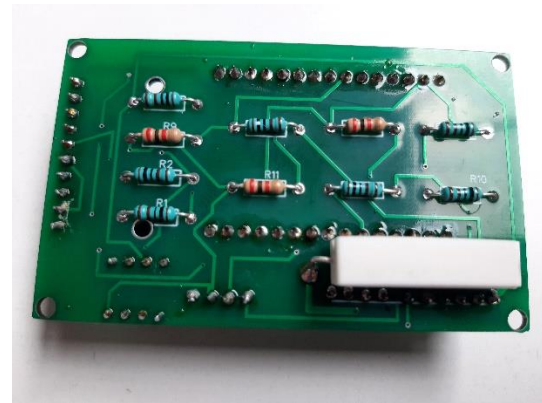
a) PCB desenvolvida, vista superior.



b) PCB desenvolvida, vista inferior.



c) PCB desenvolvida com componentes soldados, vista superior.



d) PCB desenvolvida com componentes soldados, vista inferior.

Fig. 36. Placa circuito impresso (PCB) desenvolvida com componentes soldados.

Estratégia de controlo e posicionamento

Para o controlo e posicionamento para um determinado ângulo pretendido com o motor de passo, é proposto o controlo por via de uma rampa com diferentes níveis de velocidade (número de pulsos) aplicada ao motor. O número de pulsos a aplicar ao motor de passo depende de alguns parâmetros como erro em graus e a velocidade pretendida. A Figura 37 representa o gráfico da rampa de números de pulsos aplicar em função dos parâmetros pretendidos. A rampa de controlo deve ao facto de que o tempo de ação e reação não é imediato, ou seja, o motor nunca para na

posição pretendida. O motor uma vez estando em movimento não para imediatamente devido à inércia provocada pela velocidade angular. A inércia tende a ser maior quanto maior for a velocidade. Esta técnica serve para minimizar o erro no posicionamento do motor ajustando os parâmetros para tal.

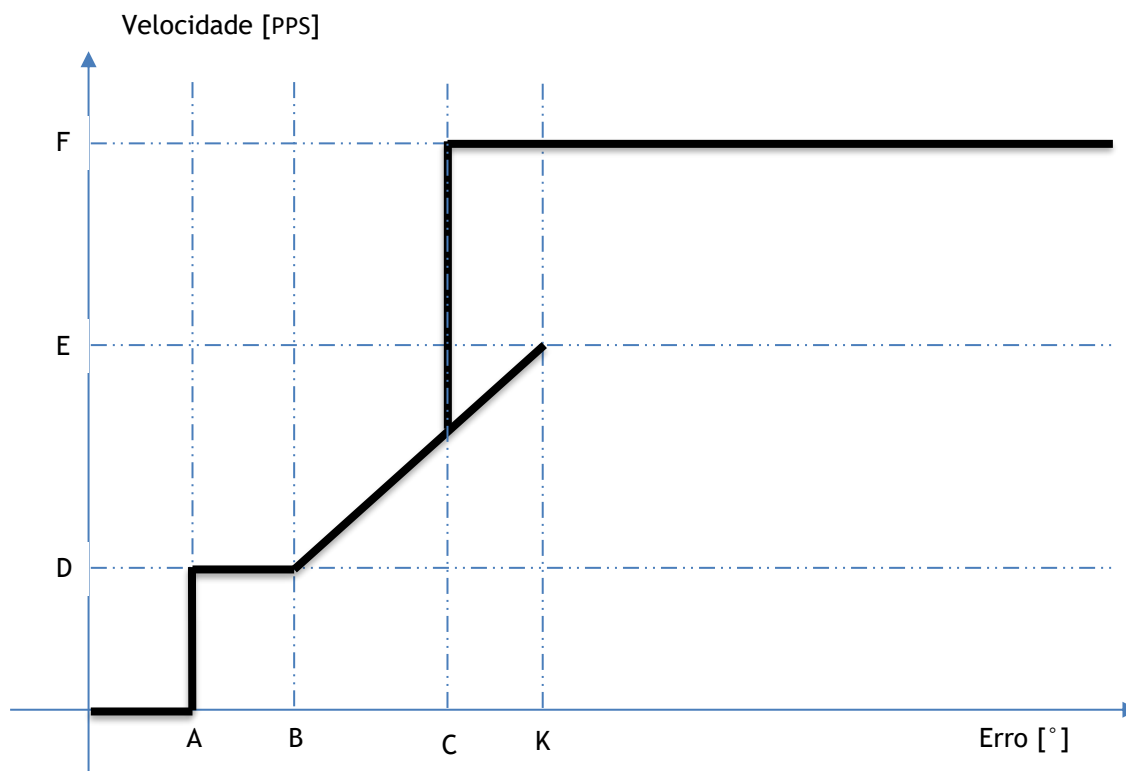


Fig. 37. Rampa de números de pulsos aplicar ao motor em função de erro.

Cujos parâmetros são:

A – Erro mínimo predefinido [°];

B – Erro mínimo para velocidade mínima constante [°];

C – Erro máximo antes de aproximação [°];

D – Velocidade mínima de aproximação pré-definida [PPS];

E – Velocidade máxima de aproximação pré-definida [PPS];

F – Velocidade de movimento fora de aproximação [PPS];

K – Erro máximo [°]

Equações de reta tangente:

$K = 2^T$, T é tipo de erro (1 – erro simples, 2 – erro quadrático) (16. a)

$C = (1 + \% \text{ velocidade normal})^T$ (eq. 16. b)

$m = \frac{E - D}{K - B}$ (eq. 16. c)

$E - D = m \cdot (K - B)$ (eq. 16. d)

$E - D = \frac{E - D}{K - B} \cdot (K - B)$ (eq. 15. e)

$Y = m \cdot x + b = \frac{E - D}{K - B} \cdot \text{Erro} - B \cdot \frac{E - D}{K - B} + D$ (eq. 16. f)

Denota-se Y como números de pulsos aplicar ao motor e é dada na Equação 15.g como:

$$Y = \begin{cases} F, & \text{se Erro} > C \\ \frac{E-D}{K-B} \cdot \text{Erro} - B \cdot \frac{E-D}{K-B} + D, & \text{se Erro} \leq C \end{cases} \quad (\text{eq. 16.g})$$

Circuito diferencial para medição de corrente elétrica do motor

Uma vez que se trata do robô colaborativo, tem de se garantir a segurança quando este colide com qualquer obstáculo, de modo a minimizar danos a ambos. Um braço robótico em colisão resulta num esforço mecânico para o motor, fazendo com que a sua corrente elétrica aumente. Para tal, foi dimensionado um circuito eletrônico com a finalidade de medir a corrente elétrica que circula no circuito da drive do motor, com a finalidade de monitorizar o seu comportamento durante o movimento do braço robótico. Esta corrente é medida com auxílio do conversor analógico digital (ADC – *Analog-to-Digital Converter*). Optou-se por utilizar um conversor em específico, ADS1115, com melhores características (menor erro de leitura, maior resolução) do que o ADC do ESP32, já que este que apresentava elevado erro de leitura e instabilidade. Na verdade, o circuito eletrônico dimensionado mede a diferença de potencial aos terminais de uma resistência conhecida e a corrente é calculada recorrendo a Lei de Ohm, uma vez que é determinado o valor da tensão elétrica aos terminais da resistência.

Formulação para dimensionamento:

A corrente máxima (I_{max}) no circuito aplicar ao motor é de 2 [A] e o ganho de ADC (ADS1115) foi configurado para +/- 2,048 V, logo o valor da resistência R5 apresentador na Fig. 34 é calculada na Equação 16.a como:

$$V_2 - V_1 = R_5 \cdot I_{max} = R_5 \cdot 2 \leq 2,048; R_5 \leq 1,0240; R_5 = 1 \Omega \text{ (eq. 16. a)}$$

A potência dissipada por efeito de Joule na resistência R_5 é dada na Equação 16.b como:

$$P_{R_5} = R_5 \cdot I_{max}^2 = 1 \cdot 2^2 \geq 4 \text{ W (eq. 16. b)}$$

A expressão geral da saída (V_o) do amplificador diferencial será dada na Equação 16.c como:

$$V_o = \left(\frac{R_2}{R_1} + 1 \right) \cdot \frac{R_4}{R_3 + R_4} \cdot V_2 - \frac{R_2}{R_1} \cdot V_1 + \frac{V_1 + V_2}{2} \cdot G_c \text{ (eq. 16. c)}$$

A Equação 16.d representa o fator de rejeição de modo comum (CMRR) como:

$$CMRR = 20 \cdot \log \left(\frac{G_d}{G_c} \right); G_c \text{ ganho modo comum; (eq. 16. d)}$$

Como R_5 é $1 [\Omega]$ e I_{max} de $2 [A]$, a variação máxima entre V_2 e V_1 será de $2 [V]$, logo o ganho diferencial $[G_d]$ é 1 . A Equação 16.e mostra simplificação para a saída (V_o) como:

$$V_o = G_d \cdot (V_2 - V_1) + \frac{V_1 + V_2}{2} \cdot G_c \text{ (eq. 16. e)}$$

Para satisfazer a condição da Equação 16.e (ganho de montagem inversora e não inversora serem iguais), a relação entre as resistências (R_1, R_2, R_3 e R_4) representado no circuito da Fig. 34 é dada na Equação 16.f como:

$$\left(\frac{R_2}{R_1} + 1 \right) \cdot \frac{R_4}{R_3 + R_4} = 1 \cap \frac{R_2}{R_1} = 1; \text{ logo } R_1 = R_2 = R_3 = R_4 \text{ (eq. 16. f)}$$

De acordo com o datasheet do amplificador operacional (TLE2141CP) utilizado, o fator de rejeição de modo comum (CMRR) é de 108 dB . A partir da Equação 16.d e com $G_d = 1$, a Equação 16.g apresenta o cálculo e condição para o ganho de modo comum (G_c) como:

$$108 = 20 \cdot \log \left(\frac{1}{G_c} \right); G_c = 4 \cdot 10^{-6}; \text{ (eq. 16. g)}$$

A expressão final da saída (V_o) é dada na Equação 16.h como:

$$V_o = 1 \cdot (V_2 - V_1) + \frac{V_1 + V_2}{2} \cdot 4 \cdot 10^{-6} \text{ (eq. 16. h)}$$

Nas condições normais $V_2 = 24V$, e $V_{1min} = 22 \text{ V}$ ($I_{max} = 2A$) e $V_{1max} = 24 \text{ V}$ ($I_{max} = 0 \text{ A}$), logo a variação da tensão de saída (V_o) aplicada ao ADC é dada pela Equação 16.i e 16.j como:

$$V_{omax} = 1 \cdot (24 - 22) + \frac{22 + 24}{2} \cdot 4 \cdot 10^{-6} = 2,00009 \text{ V} \leq 2,048 \text{ V (eq. 16. i)}$$

$$V_{omin} = 1 \cdot (24 - 24) + \frac{24 + 24}{2} \cdot 4 \cdot 10^{-6} = 0,096 \text{ mV} \text{ (eq. 16. j)}$$

Divisor de tensão (5 V/3,3 V)

Como as entradas/saídas do ESP32 funcionam a uma tensão de 3,3 V, e o sinal do encoder (A+ e B+) e do TIC 500 (Tx) são a 5 V, é preciso aplicar um divisor de tensão às Equações 16.k e 16.l para calcular os valores das resistências e na Equação 16.m a potência dissipada (P_{R7} e P_{R6}).

$$V4 = \frac{R7}{R6 + R7} \cdot V3; \quad V3 = 5V; \quad V4 = 3,3V; \text{ (eq. 16. k)}$$

$$3,3 = \frac{R7}{R6 + R7} \cdot 5 \Leftrightarrow R7 = 1,95 \cdot R6 \Leftrightarrow R7 = 2 \cdot R6 \text{ (eq. 16. l)}$$

$$P_{R7} = \frac{V_{R7}^2}{R7} = \frac{3,3^2}{2000} = 5,45 \text{ mW}; \quad P_{R6} = \frac{V_{R6}^2}{R7} = \frac{(5 - 3,3)^2}{1000} = 2,89 \text{ mW}; \text{ (eq. 16. m)}$$

A Tabela 12 apresenta os valores das resistências, potência dissipada e tolerância das resistências elétrica utilizado no circuito da Figura 35.

Tabela 13. Valores das resistências utilizada.

Resistências [Ω]	Potencia [W]	Tolerâncias [%]
R1	18 K	1/4
R2	18 K	1/4
R3	18 K	1/4
R4	18 K	1/4
R5	1	5
R6	1 K	1/4
R7	2 K	1/4
R8	1 K	1/4
R9	2 K	1/4
R10	1 K	1/4
R11	2 K	1/4

Cálculo da velocidade

Para calcular o número de pulsos a aplicar ao motor de passo dependendo do tipo de passo, recorre-se ao datasheet de cada motor, obtendo a velocidade máxima e ângulo por passo. Alguns parâmetros foram calculados como:

- Número de pulsos por volta [PPV] referente ao passo completo representado na Equação 17.a como:

$$PPV = \frac{360}{1,8} = 200 \text{ pulsos} \quad (\text{eq. 17. a})$$

- Pulso Por Segundo [PPS] a passo completo representado na equação 17.b:

$$PPS = \text{Pulso por volta [PPV]} \cdot \text{Velocidade máxima [RPS]} \quad (\text{eq. 17. b})$$

- Pulso Por Segundo [PPS] a 1/2 passo representado na equação 17.c:

$$PPS = \text{Pulso por volta [PPV]} \cdot \text{Velocidade máxima [RPS]} \cdot 2 \quad (\text{eq. 17. c})$$

- Pulso Por Segundo [PPS] a 1/4 passo representado na equação 17.d:

$$PPS = \text{Pulso por volta [PPV]} \cdot \text{Velocidade máxima [RPS]} \cdot 4 \quad (\text{eq. 17. d})$$

- Pulso Por Segundo [PPS] a 1/8 passo representado na equação 17.e:

$$PPS = \text{Pulso por volta [PPV]} \cdot \text{Velocidade máxima [RPS]} \cdot 8 \quad (\text{eq. 17. e})$$

A Tabela 13 apresenta os valores calculados da velocidade (Pulso Por Segundo [PPS]) a aplicar ao motor para diferentes tipos de passo.

Tabela 14. Parâmetro velocidade (Pulso Por Segundo [PPS]) aplicar aos motores.

	Eixo	Ângulo completo [°/passo]	Pulso por volta [PPV]	Velocidade máxima [RPS]	Pulso Por Segundo [PPS]			
					Passo completo	1/2 passo	1/4 passo	1/8 passo
Motor de Passo	1	1,8	200	7	1400	2800	5600	11200
	2	1,8	200	6	1200	2400	4800	9600
	3	1,8	200	11	2200	4400	8800	17600
	4	1,8	200	11	2200	4400	8800	17600
	5	1,8	200	11	2200	4400	8800	17600
	6	1,8	200	11	2200	4400	8800	17600

Conversão de pulsos em graus

Para calcular a posição em graus através de número de pulsos do encoder, a resolução do encoder [PPR] foi programada para 2500 pulsos por voltas. O sistema engrenagem designado por planetário constituído por: pinhão solar, planetário e coroa/anel (externo) tem: o pinhão abreviado por P tem 15 dentes, planetário na parte inferior (Pi) tem 18 dentes e superior (Ps) tem 15 dentes e o anel exterior (Ae) tem 48 dentes. A relação da caixa planetária (RCP) é calculada na Equação 18.a como:

$$RCP = \frac{n^{\circ} \text{ dentes P}}{n^{\circ} \text{ dentes Pi}} \cdot \frac{n^{\circ} \text{ dentes Pi}}{n^{\circ} \text{ dentes Ps}} \cdot \frac{n^{\circ} \text{ dentes Ps}}{n^{\circ} \text{ dentes Ae}} = \frac{15}{18} \cdot \frac{18}{15} \cdot \frac{15}{48} = 0,3125 \text{ volta} \quad (\text{eq. 18. a})$$

O resultado da Equação 18.a representa a volta em graus do elo quando o veio do motor de passo efetua uma volta completa, ou seja, quando o motor dá uma volta (360 graus) completa, o elo do braço efetua 0,3125 volta (112,50°).

O ângulo por pulso completo do encoder em relação ao eixo do motor APCEM (resolução em relação ao veio do motor) é calculado na Equação 18.b como:

$$APCEM = \frac{360}{2 \cdot PPR} = \frac{360}{2 \cdot 4096} = 0,0439 [^{\circ}] \quad (\text{eq. 18. b})$$

O ângulo por pulso completo do encoder em relação ao elo do braço APCEE (resolução em relação ao elo do braço robótico) é calculado na Equação 18.c como:

$$APCEE = APCEM \cdot RCP = 0,0439 \cdot 0,3125 = 0,0137 [^\circ] \text{ (eq. 18. c)}$$

Treino da rede neuronal artificial

A colisão do braço robótico com algum obstáculo resulta de um esforço mecânico no motor e que por sua vez faz variar a corrente elétrica. A rede neuronal foi encarregue de detetar essas possíveis variações de corrente com base nos dados dos testes feitos. Para aquisição de dados para o treino da rede, foram feitos vários testes com braço robótico em que geravam as posições e velocidade aleatórias com diferentes níveis de carga acoplado ao efector e registava-se a corrente máxima atingida.

Os dados de entrada para treino de rede são: posição pretendida, erro associado, carga aplicada ao efector, velocidade e tipo de passo a cada movimento e como saída a corrente máxima estimada. Recorrendo ao programa de Matlab para o treino, validação e teste da rede neuronal artificial, foram consideradas 1000 iterações, 100 neurónios escondido para entrada e 3 para saída e algoritmo de treino de “Levenberg Marquardt”. O algoritmo Levenberg Marquard (Ngia & Sjöberg, 2000), é um algoritmo para a aprendizagem de redes neuronais artificiais que avalia a primeira derivada da função de custo em relação aos pesos. Trata-se de um algoritmo eficiente para adaptar sistemas não lineares, tanto em dados simulados como em dados práticos reais, apresentando uma elevada eficiência na convergência (Rubio, 2020).

A Figura 38 representa a estrutura de rede neuronal artificial criada e na Tabela 14 apresentam-se os números das amostras seleccionadas, erro médio quadrático (RSE) e a respetivo valor de regressão (R) linear.

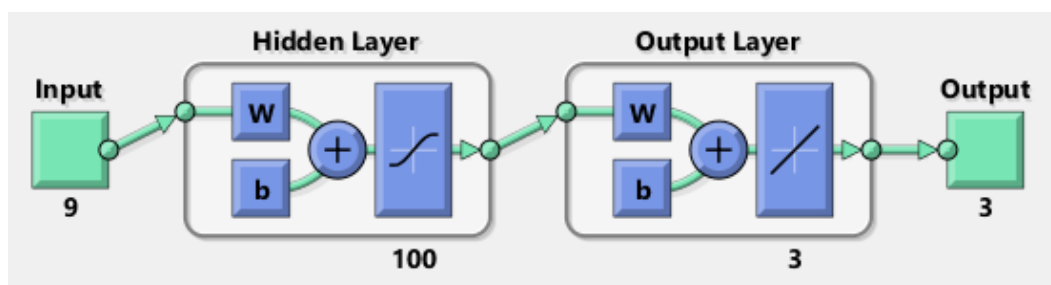


Fig. 38. Estrutura de rede neuronal artificial criada.

Tabela 15. Resultado de treino da rede neuronal artificial.

	Percentagens das amostras	MSE	R
Treino	70	6826,776E-0	8,8435E-0
Validação	15	7520,1074E-0	9,293E-01
Teste	15	20599,486E-0	8,430E-01

A Figura 39 representa o desenvolvimento, ou seja, a evolução dos erros durante o processo de treino da rede.

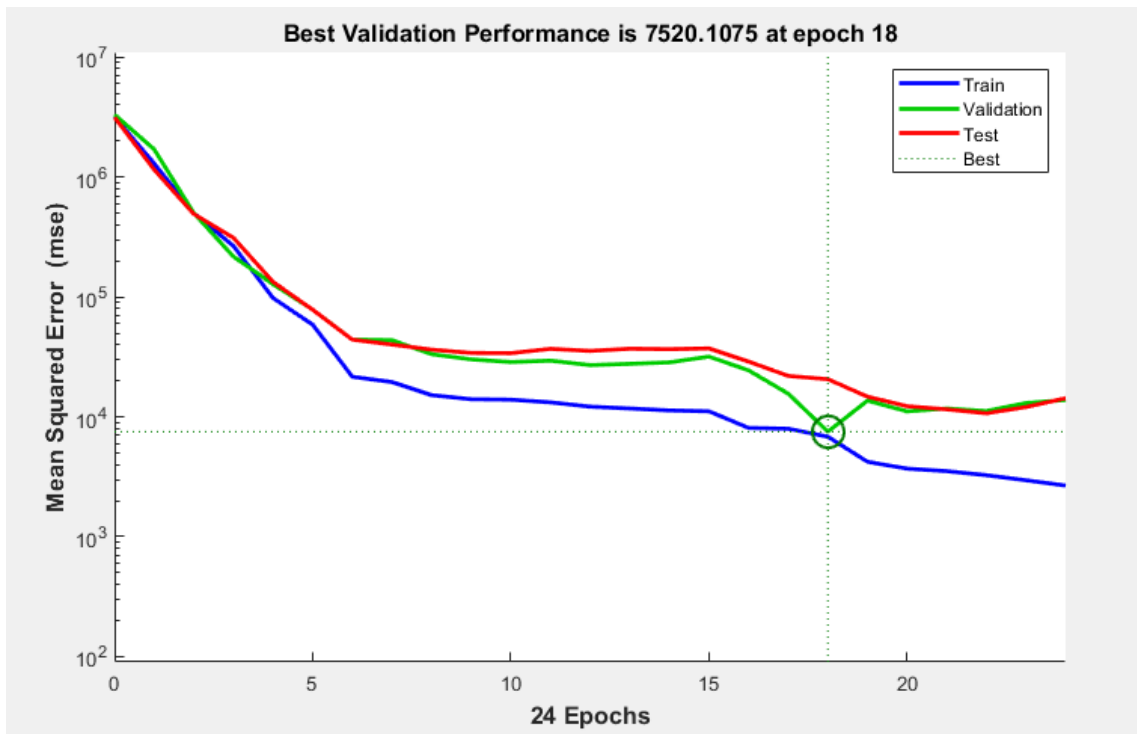


Fig. 39. Evolução do treinamento da rede neuronal com o número de iteração.

A rede neuronal artificial prevê o resultados através do treino base nos dados amostras. Estes resultados tendem a ser melhores e mais assertivos quanto maior for o número de dados das amostras. Dos testes realizado foram selecionadas aleatoriamente 100 dados, comparando os valores da corrente medida e estimada pela rede, assim como o erro associado representado na Figura 40.

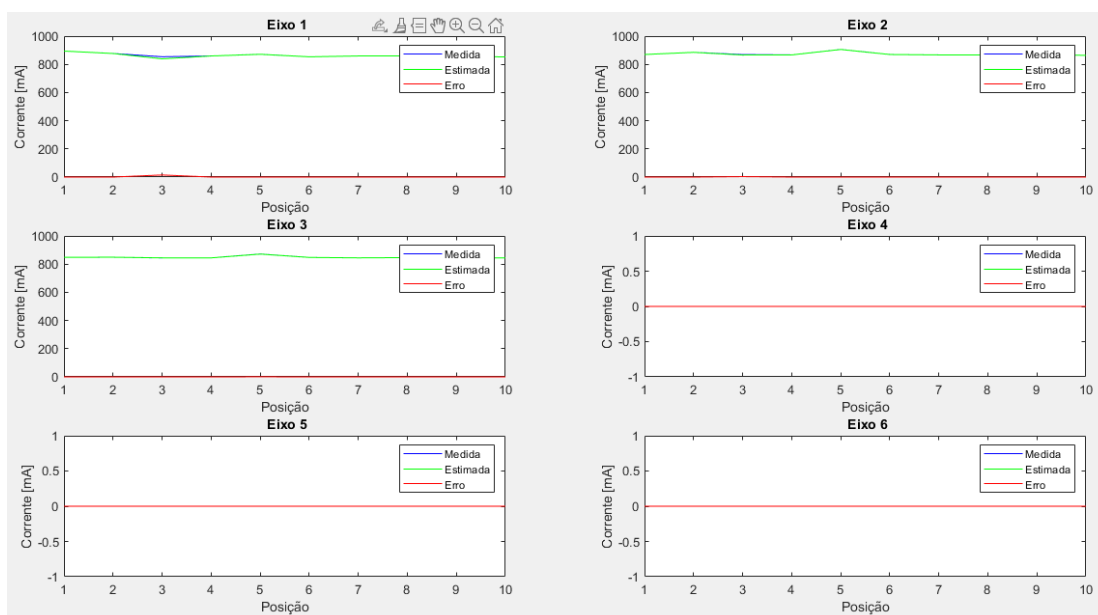


Fig. 40. Comparação entre corrente medida e estimada pela rede neuronal.

4.4. Interface gráfico do utilizador - MATLAB

Para estabelecer a comunicação entre humano e o braço robótico, foi desenvolvido uma interface gráfica do utilizador (GUIDE - *Graphical User Interface Development Environment*) no Matlab com diversos menus (menu conectar, alarme, controlo e programa) e sendo possível estabelecer a comunicação, visualizar ou alterar os parâmetros e movimentar o braço robótico.

O menu designado “conectar” representado na Figura 41 (lado esquerdo superior), permite estabelecer a comunicação com o braço robótico selecionando a porta de comunicação (Port) e a taxa de transmissão (Baud Rate), estando configurado para 500000 bits por segundo. Esta janela permite visualizar e alterar os parâmetros de cinemática inversa, comprimentos dos elos, limite de movimento de cada eixo e distâncias de colisão. Permite ainda, enviar e receber comandos. No lado direito da Figura 41, designado como comando stand, aparece em todos os menus, sendo possível sinalizar alarme, comunicação, estado da drive e também permite selecionar o modo de execução (manual, automático ou teste), habilitar, desabilitar e rearmar os defeitos do robô e também permite enviar alguns comandos predefinidos.



Fig. 41. Interface do utilizador, menu “Conectar”

A Figura 42 representa o menu designado “Mover”, sendo que neste menu é possível visualizar a posição em graus de cada junta, posição e orientação da ferramenta, efetuar o movimento de cada juntas, mover/orientar a ferramenta e bem como alterar a percentagem da velocidade de movimento. Também é possível mudar o valor incremental da percentagem de velocidade, do ângulo para juntas e a distância para ferramenta. Por fim, é possível movimentar o braço robótico para a posição de calibração (posição inicial referenciada).

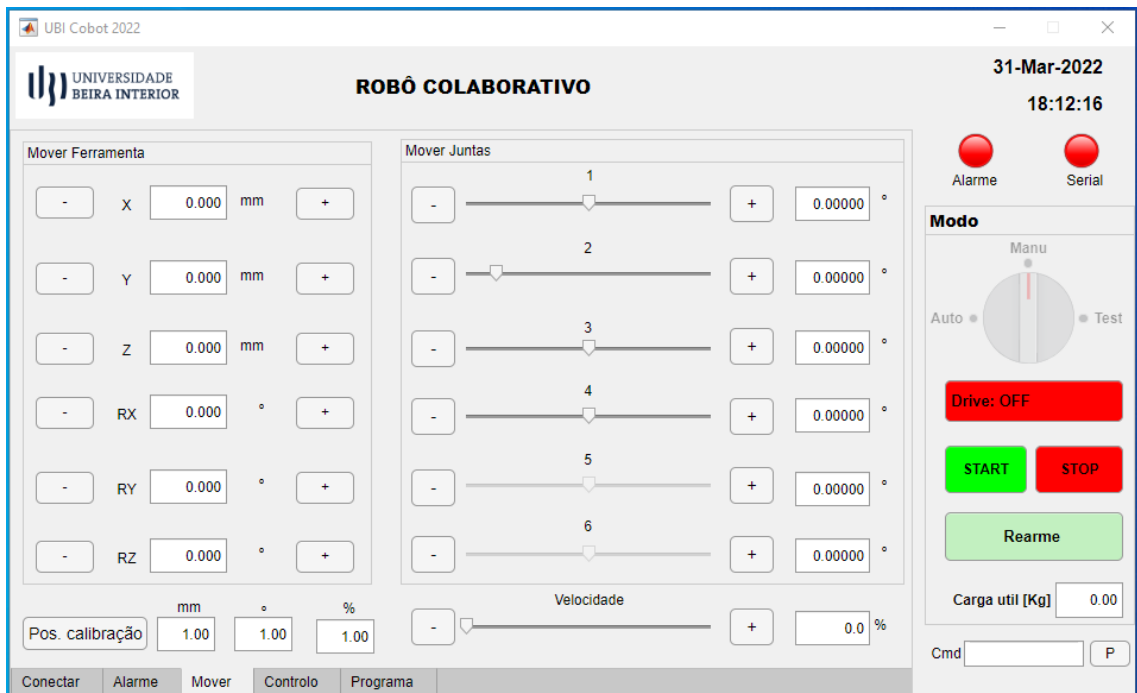


Fig. 42. Interface do utilizador, menu “Mover”.

O menu “Alarme” representado na Figura 43 permite visualizar o(s) alarme(s) ativo(s) e bem como o histórico dos alarmes. Estes alarmes são notificados com a data e hora da ocorrência e a descrição do mesmo.

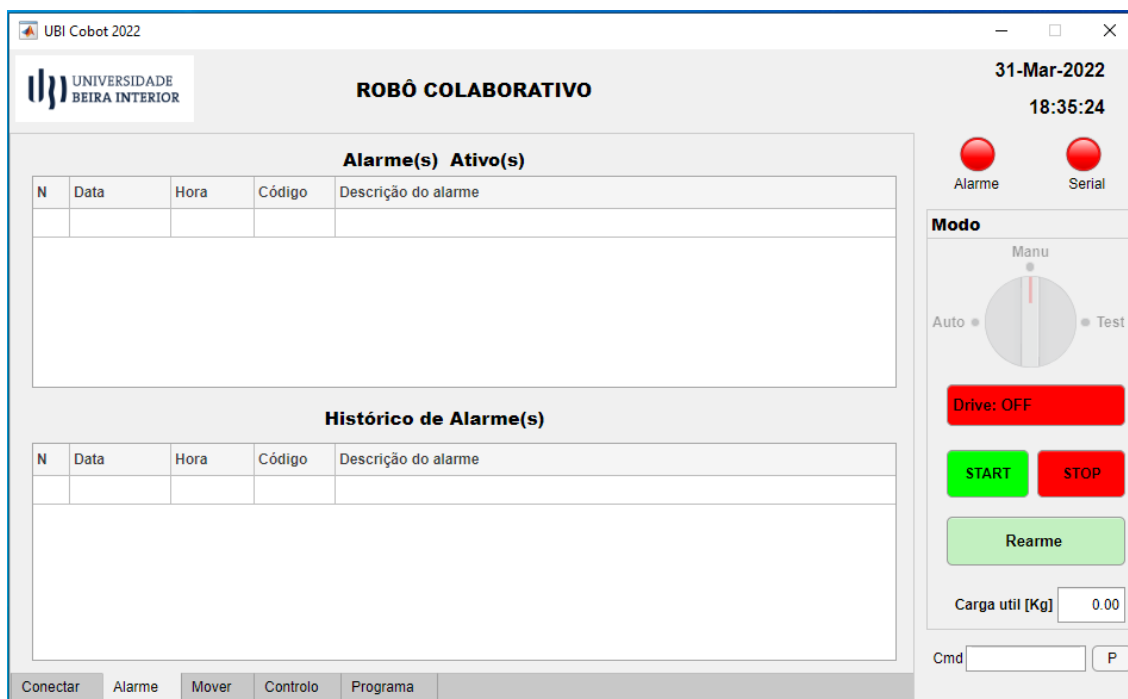


Fig. 43. Interface do utilizador, menu “Alarme”.

No menu designado por “Controlo”, representado na Figura 44, permite visualizar alguns dados como a tensão aplicada ao driver do motor, a corrente consumida e o(s) de pulso(s) do encoder. É possível visualizar e alterar o estado de driver, calibrar os eixos, reiniciar o(s) ESP32s, alterar tipo de passo a movimentar e também alterar e visualizar os parâmetros de controlo.

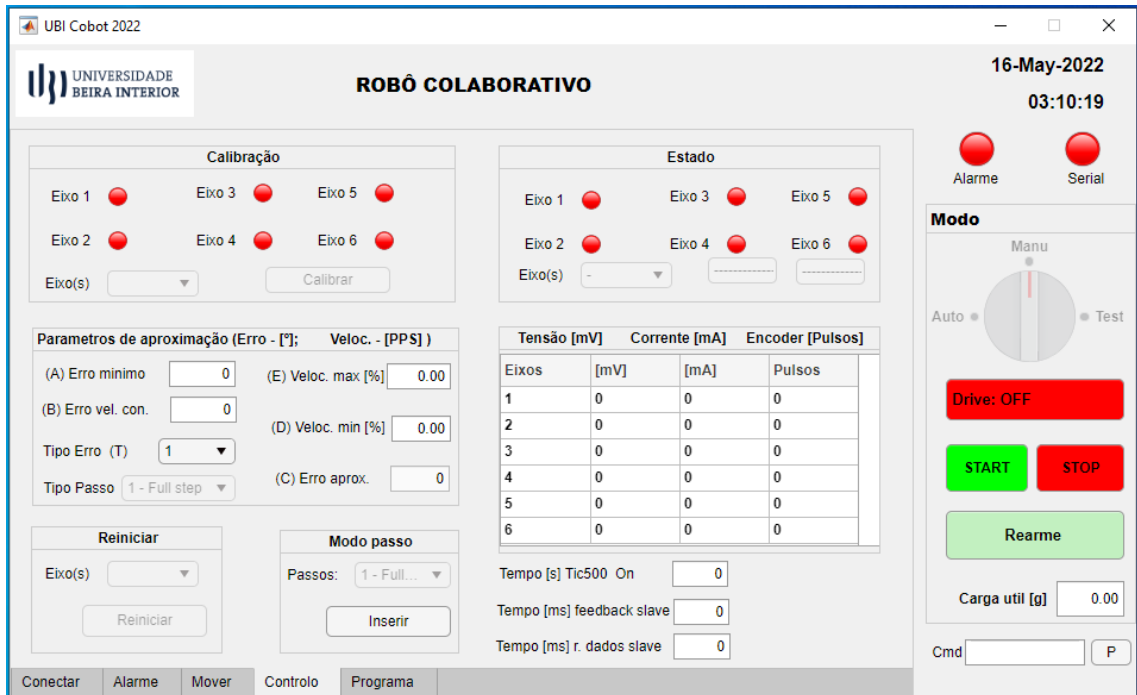


Fig. 44. Interface do utilizador, menu “Controlo”.

De modo para o robô executar uma sequência de movimentos (pontos), foi criado um último menu designado por “Programa” representado na Figura 45, sendo possível criar, alterar, executar e apagar pontos. Esta janela permite incluir uma sequência de movimentos e gravá-la, tal que o robô, repita essa sequência o número de vezes programado, ou então, indefinidamente.

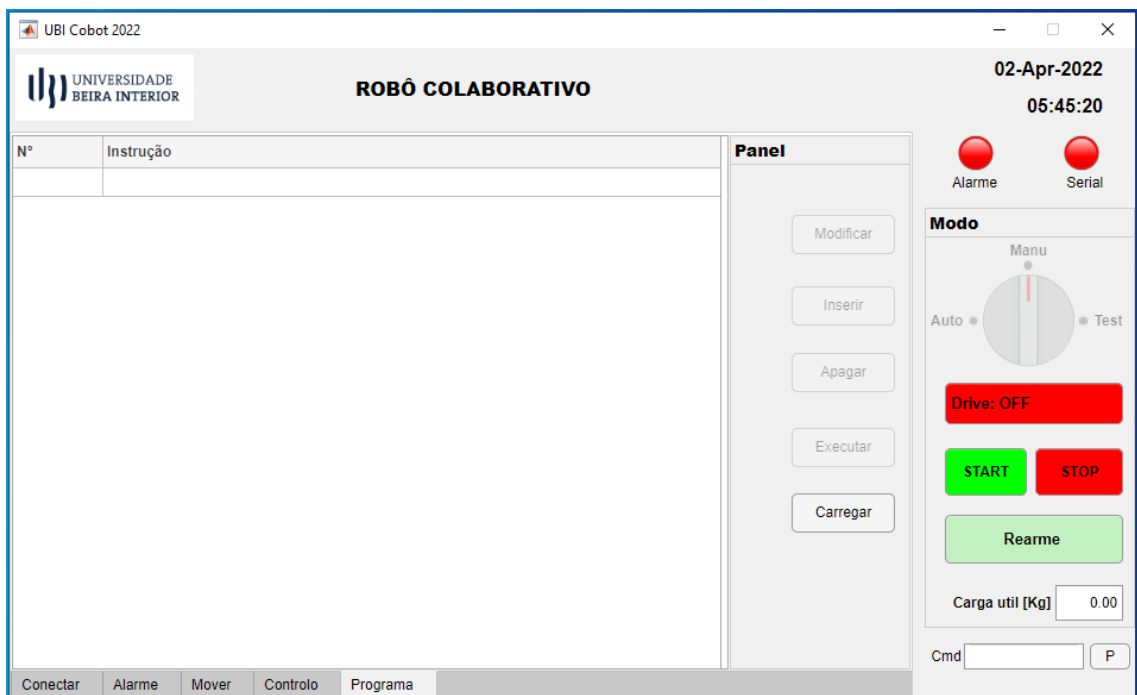


Fig. 45. Interface do utilizador, menu “Programa”.

4.5. Impressões 3D e montagem do braço robótico

Após ter as partes do desenho 3D desenvolvidas no software SolidWorks, as mesmas foram gravadas em extensão “stl” (arquivo estereolitografia) e posteriormente convertidas para extensão GCODE (código G) reconhecível na impressora 3D. As peças foram impressas por uma impressora 3D (Artillery Sidewinder x1) maioritariamente com filamento PLA e PETG. Na Tabela 15 encontram-se representados os parâmetros ideais configuráveis para diferentes tipos de filamento para processo de impressão.

Tabela 16. Parâmetros principais de impressão para diferente filamento.

Filamento	Temperatura [°C]		Velocidade impressão [mm/s]	Altura da camada[mm]	Densidade de infiltração [%]
	Impressão	cama			
PLA	215	70	80	0,2	20 - 50
PETG	240	80	80	0,2	40

A Figura 46 apresenta o processo de evolução na montagem do braço robótico.



a) Fase inicial, montagem da base.

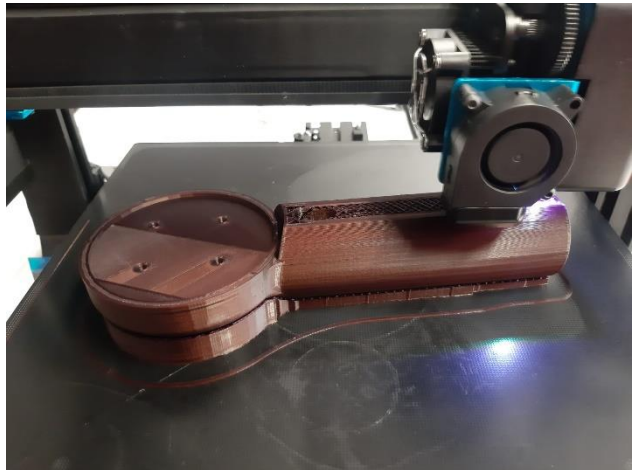


b) Montagem e estrutura do eixo 2.

Fig. 46. Impressões e processo evolutivo da montagem do braço robótico.



c) Montagem pinhões e motor do eixo 2.



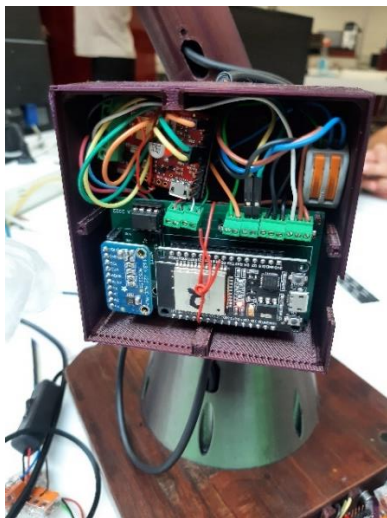
d) Impressão do ombro de elevação do eixo 2.



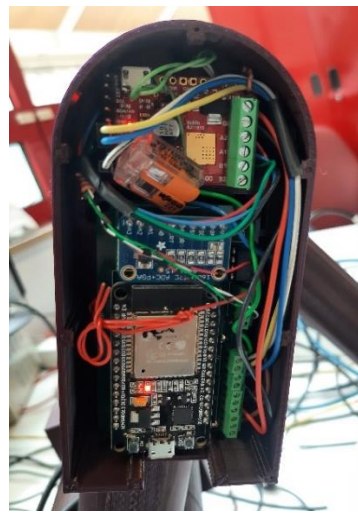
e) Ombro de elevação para eixo 3.



f) Montagem final dos 3 primeiros eixos.

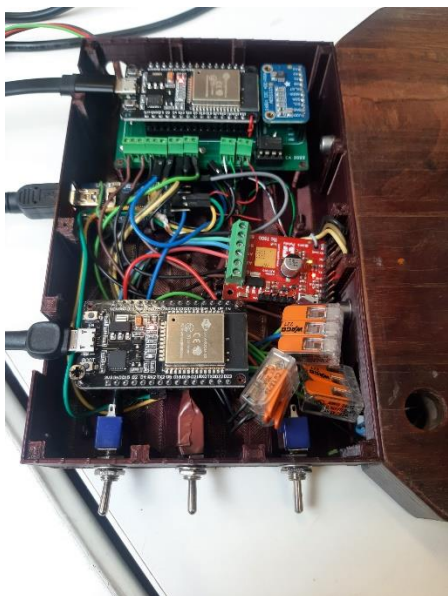


g) Ligação dos componentes de eixo 2

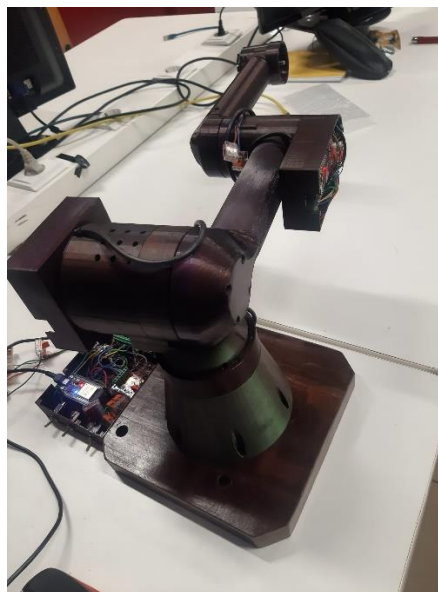


h) Caixa com ligação dos componentes de eixo 3

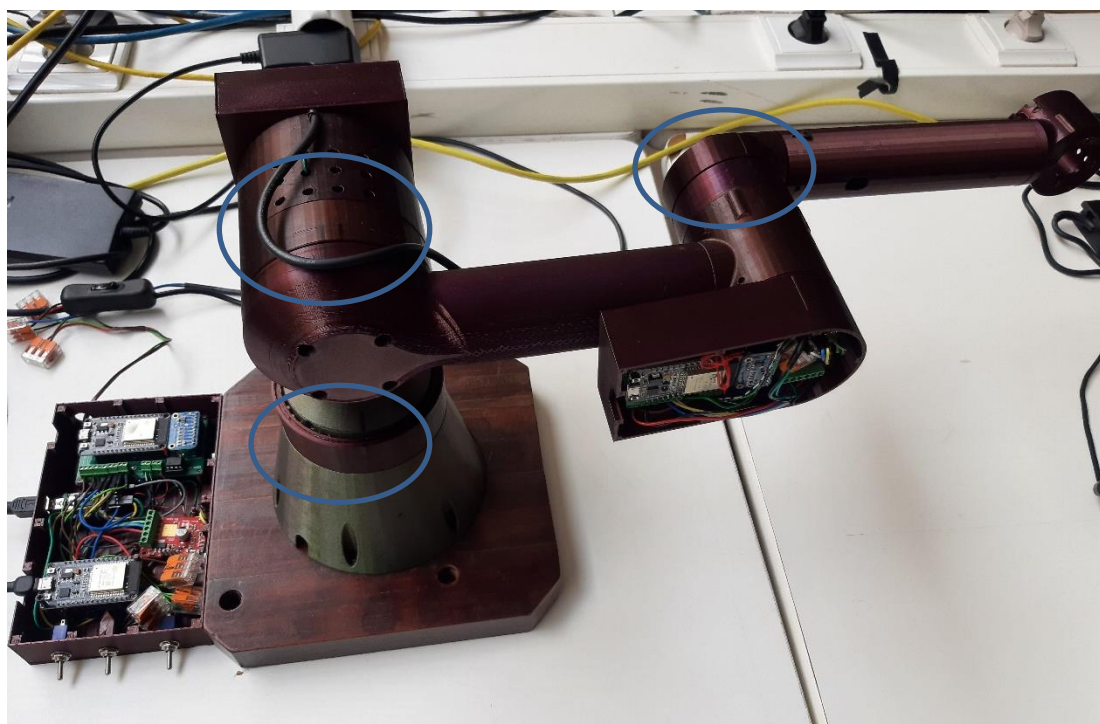
Fig. 46. Impressões e processo evolutivo da montagem do braço robótico (cont.).



i) Caixa com ligação dos componentes de eixo 1 e ESP32 principal.

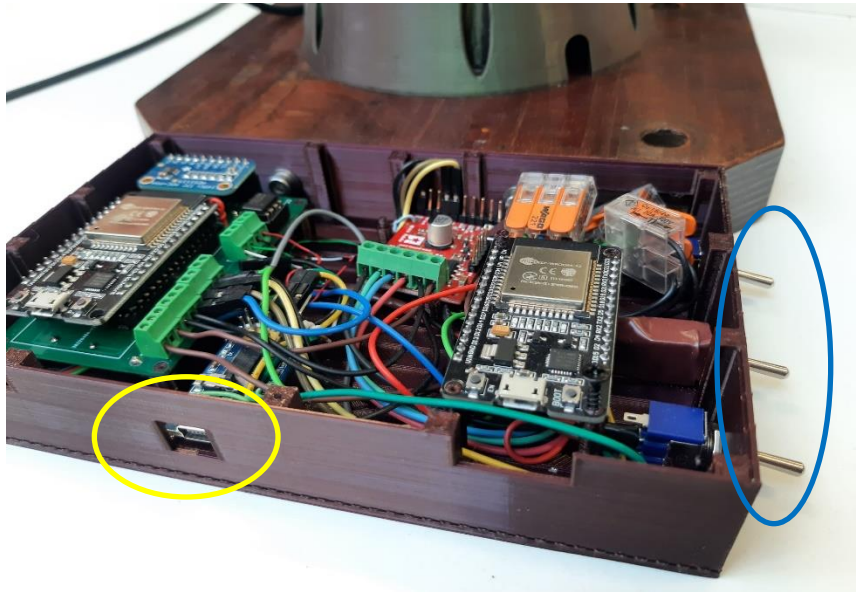


j) Vista cima da montagem final dos 3 primeiros eixos.



k) Vista cima da montagem final destacando as 3 juntas para cada eixo.

Fig. 46. Impressões e processo evolutivo da montagem do braço robótico (cont.).



m) Switch (azul) On/Off ESP32 e drive e entrada USB (amarelo) interligar robô e PC - Matlab



m) Montagem final, localização das eletrônica para a 3 juntas.

Fig. 46. Impressões e processo evolutivo da montagem do braço robótico (cont.).

4.6. Nota conclusiva

Ao longo deste capítulo foram descritas todas as etapas até à montagem final do braço robótico, passando pela formulação da ideia e adaptação do desenho original, o dimensionamento, controlo, impressão a montagem e, por fim, os testes.

Inicialmente adaptou-se o desenho aos requisitos definidos, alterando as partes do desenho e posteriormente procedeu-se à impressão 3D. Este procedimento auxiliou no processo de seleção dos motores para estrutura robótica. Posto isso, avançou-se pensando em método, estratégias de controlo e interface entre o braço robótico e o utilizador. Por fim, foi realizada a montagem final do braço, tendo em consideração eventuais barreiras que foram surgindo até ao testes e aquisição de dados.

5. Análise de Discussão de Resultados

Neste capítulo são apresentados os resultados dos testes realizados ao robô colaborativo desenvolvido. Os testes realizados foram relacionados com a carga útil e com a precisão do posicionamento.

5.1. Teste N.º 1: Carga útil

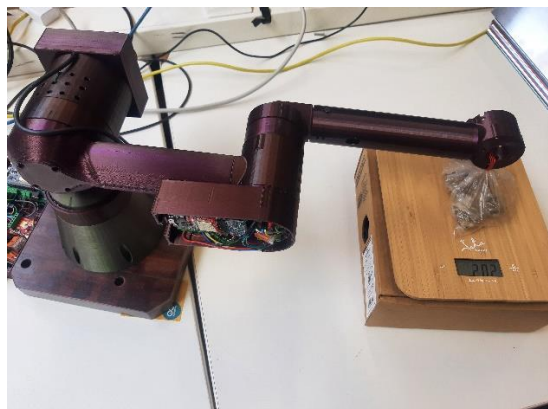
O Teste n.º 1 residiu na determinação da carga útil, i.e., o peso máximo que o manipulador robótico suporta tanto parado e em movimento. Quando os motores do manipulador não estão acionados, a carga é suportada pelo binário criado na caixa redutora. Quando em movimento, é assegurada pelo binário produzido pelo(s) motor(es). Para efetuar o teste mencionado recorre-se à consideração física em que a força é máxima para um determinado peso, quanto maior for a distância horizontal (0°) da massa ao centro do movimento. Para tal, o braço robótico foi estendido na horizontal, o que corresponde a zero graus (0°) para os eixos 2 e 3 e foi-se adicionando peso gradualmente no centro do efector final até o braço começar a descair. Posteriormente pesaram-se as massas, obtendo um peso máximo de 314 g. Os pesos graduais encontram-se na Figura 47 no decorrer dos testes. A Figura 47 representa várias etapas do teste n.º 1 e o resultado obtido.



a) Teste n.º 1, robô em posição sem carga.



b) Teste n.º 1, robô em posição com 108g de carga.



c) Teste 1, robô em posição com 202g de carga.



d) Teste 1, robô em posição com 262g de carga.



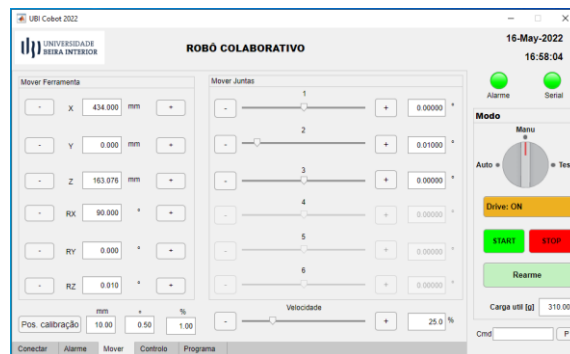
e) Teste 1, robô em posição com 314g de carga.



f) Pesagem final da carga útil (314g).



g) Teste 1, robô em posição com a carga útil (314g) acoplado



i) Menu mover, dados da posição, orientação do efeetor e ângulo de cada juntas.

Fig. 47. Robô posicionado para teste n.º 1 de carga útil.

O restado do teste n.º 1, carga útil, permite determinar o peso máximo que o braço robótico suporta tanto parado ou em movimento e com ou sem o braço todo estendido. Comparando o resultado com o robô UR3 da Universal Robot que se pretendeu replicar, suporta 3 kg, o robô desenvolvido suporta apenas 1 décimo do peso, 0,314 Kg. Este resultado é bastante aceitável, uma vez que braço desenvolvido é todo feito em PLA, incluindo a pinhão e o conjunto de engrenagens, motor sem bloqueio mecânico e de baixo binário eletromecânico.

5.2. Teste N.º 2: Erro de posicionamento

O Teste n.º 2 residiu na avaliação da precisão do posicionamento, de forma avaliar o método e estratégia da Figura 36, pelo que requereu que fossem selecionadas 5 posições. Posicionou-se automaticamente o robô a 35% de velocidade e com a carga útil acoplada. Registaram-se as variações nas posições antes e após desabilitar o driver do motor. A Tabela 16 apresenta os ângulos de cada junta, posição e orientação das 5 posições selecionadas para o teste n.º 2. A Tabela 17 apresenta os resultados de teste n.º 2 e a Tabela 18 apresenta o erro médio quadrático influenciado pelo teste 2 em relação aos dados da Tabela 16. A Figura 48 representa o robô em diferentes posições durante o teste n.º 2. Estas sequências de movimentos servem para averiguar se o manipulador consegue movimentar-se com a carga útil do teste n.º 1 e também para saber qual a postura do mesmo em relação aos dados da Tabela 16.



a) Teste n.º 2, robô em posição P0 (origem).



b) Teste n.º 2, robô em posição P1.



c) Teste n.º 2, robô em posição P2.



d) Teste n.º 2, robô em posição P3.

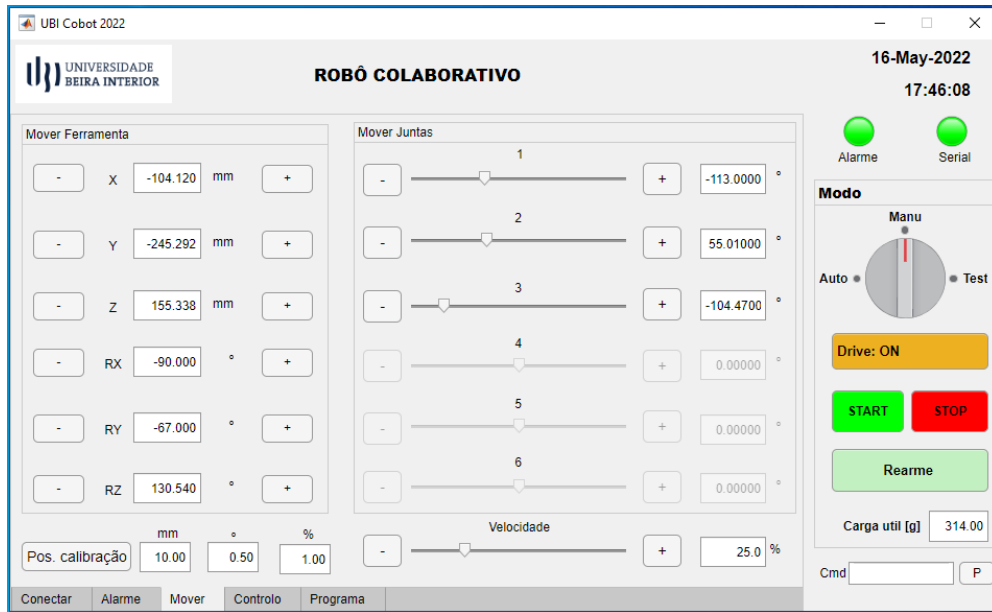


e) Teste n.º 2, robô em posição P4.

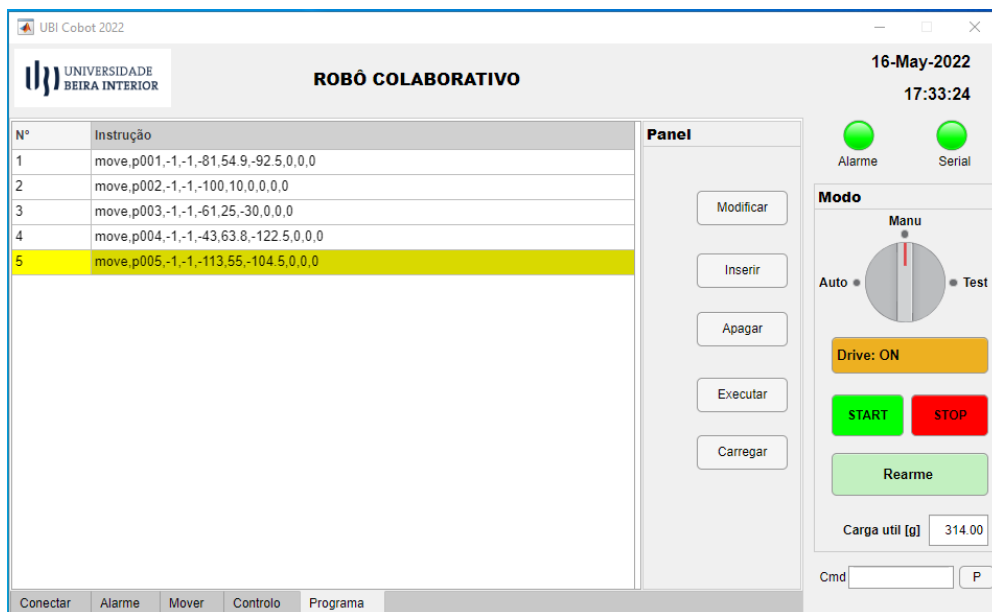


f) Teste n.º 2, robô em posição P5.

Fig. 48. Robô em diferentes posições durante o teste n.º 2 no posicionamento.



g) Teste 2, posição, orientação e a ângulo das juntas do robô na posição P5



i) Programa executado durante o teste 2.

Fig. 48. Robô em diferentes posições durante o teste n.º 2 no posicionamento (cont.).

Tabela 17. Ângulos, posições e orientações escolhido para teste n.º 2 de posicionamento.

Posição	Ângulos das juntas [°]			Posição [mm]			Orientação [°]		
	θ_1	θ_2	θ_3	X	Y	Z	X	Y	Z
P1	-81,0	54,9	-92,5	46,8	-295,8	189,6	90,0	-81,0	-37,6
P2	-100,0	10,0	0,0	-74,2	-420,9	238,4	-90,0	-80,0	-170,0
P3	-61,0	25,0	-30,0	200,7	-362,1	229,2	90,0	-61,0	-5,0
P4	-43,0	63,8	-122,5	153,3	-142,9	149,5	90,0	-43,0	-58,7
P5	-113,0	55,0	-104,5	-104,1	-245,2	155,2	-90,0	-67,0	130,5

Tabela 18. Ângulos, posições e orientações, resultado do teste n.º 2 de posicionamento.

Movimentos mantendo drive habilitado						
		P1	P2	P3	P4	P5
Ângulos [°]	θ_1	-81,01	-99,99	-61,00	-43,00	-112,99
	θ_2	54,90	10,01	25,01	63,79	54,99
	θ_3	-92,56	0,02	-29,98	-122,49	-104,52
Posição [mm]	X	46,78	-74,14	200,71	153,28	-104,02
	Y	-295,70	-420,90	-362,10	-142,93	-245,17
	Z	189,38	238,52	229,32	149,50	155,12
Orientação [°]	X	90,00	-90,00	90,00	90,00	-90,00
	Y	-81,01	-80,00	-61,00	-43,00	-67,01
	Z	-37,66	-169,97	-4,97	-58,70	130,47
Movimentos mantendo drive desabilitado						
		P1	P2	P3	P4	P5
Ângulos [°]	θ_1	81,01	-99,99	-61,01	-43,00	-112,99
	θ_2	54,90	9,91	24,93	63,79	54,94
	θ_3	-92,57	-0,10	-29,94	-122,53	-104,56
Posição [mm]	X	46,78	-74,18	200,70	153,18	-104,63
	Y	295,68	-421,11	-362,23	-142,85	-246,62
	Z	189,35	237,30	228,90	149,42	156,24
Orientação [°]	X	90,00	-90,00	90,00	90,00	-90,00
	Y	81,00	-80,01	-61,01	-43,00	-67,01
	Z	-37,67	-170,19	-5,01	-58,74	130,94

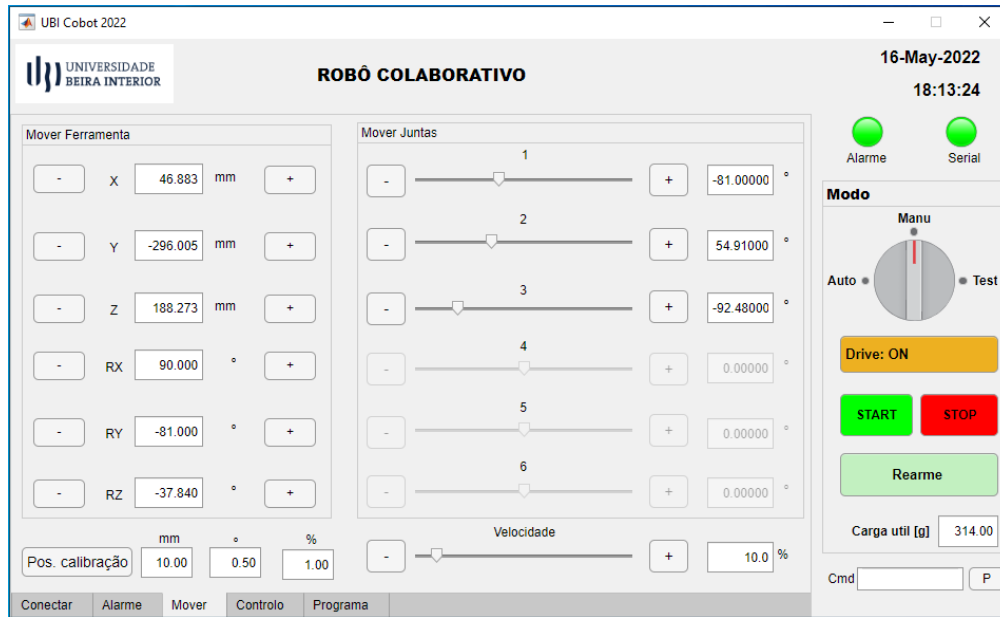
Tabela 19. Erro médio quadrático associado ao teste n.º 2.

Erro médio quadrático (RMSE)					
	Movimentos mantendo driver habilitado				
	P1	P2	P3	P4	P5
Juntas [°]	$3,51 \times 10^{-2}$	$1,41 \times 10^{-2}$	$1,29 \times 10^{-2}$	$8,16 \times 10^{-3}$	$1,41 \times 10^{-2}$
Posição [mm]	$1,42 \times 10^{-1}$	$9,99 \times 10^{-2}$	$8,80 \times 10^{-2}$	$1,78 \times 10^{-2}$	$7,25 \times 10^{-2}$
Orientação [°]	$3,51 \times 10^{-2}$	$1,73 \times 10^{-2}$	$1,73 \times 10^{-2}$	0	$1,83 \times 10^{-2}$
	Movimentos mantendo driver desabilitado				
	P1	P2	P3	P4	P5
Juntas [°]	$4,08 \times 10^{-2}$	$7,79 \times 10^{-2}$	$5,35 \times 10^{-2}$	$1,83 \times 10^{-2}$	$4,93 \times 10^{-2}$
Posição [mm]	$1,65 \times 10^{-1}$	$6,27 \times 10^{-1}$	$1,71 \times 10^{-1}$	$8,44 \times 10^{-2}$	1,06
Orientação [°]	$4,04 \times 10^{-2}$	$1,10 \times 10^{-1}$	$8,16 \times 10^{-3}$	$2,31 \times 10^{-2}$	$2,54 \times 10^{-1}$

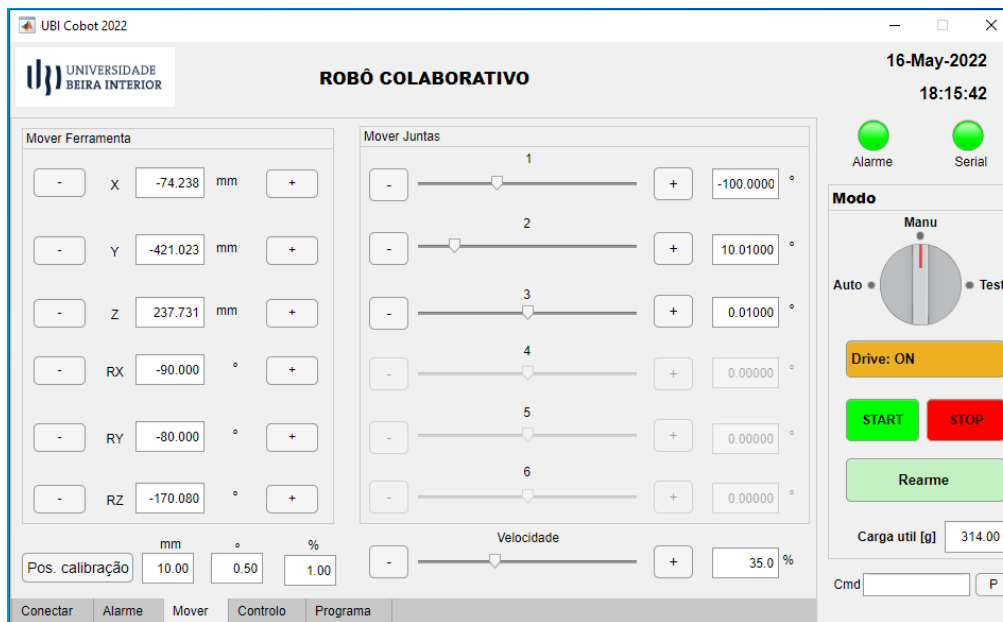
Os resultados obtidos no teste n.º 2, compilados na Tabela 17, comparando com os resultados da Tabela 16, facilmente permitem observar que o posicionamento apresenta um erro (Tabela 18) relativamente baixo e aceitável, sendo resultado de desgaste do pinhão e do anel dentado, erro na medição e conversão, tempo de ação e resposta ao motor e também a da inércia provocada pelo peso da carga útil e o braço robótico.

5.3. Teste N.º 3: Precisão e repetibilidade

O teste n.º 3 residiu na avaliação da precisão e a repetibilidade de acordo com ISO 9283 (International Organisation of Standardization, 1998). Este teste resulta em 30 movimentos repetitivos percorrendo sempre uma sequência de 5 posições representadas na Figura 50. Este teste foi feito com o máximo de carga útil, temperatura ambiental de 20 °C e com a velocidade a 10 e 35% da velocidade. As posições (P1 a P5) para o teste de repetibilidade são iguais à do teste 2, exceto P4. Na Figura 49 á 52 encontram-se representado o braço robótico na posição P4 e os dados da posição e orientação em cada posição durante o teste de precisão e repetibilidade. Estas figuras servem para demonstrar que o manipulador atingiu as posições, as quais podem ser confirmadas na Tabela 20, 21 e 22.

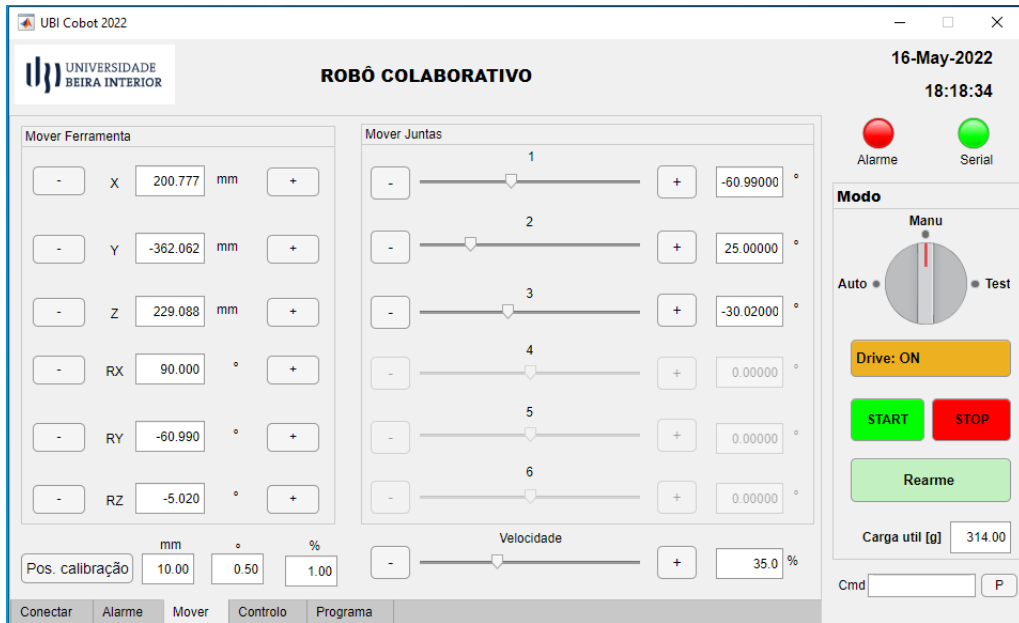


a) Teste n.º 3, posição, orientação e a ângulo das juntas do robô na posição P1.

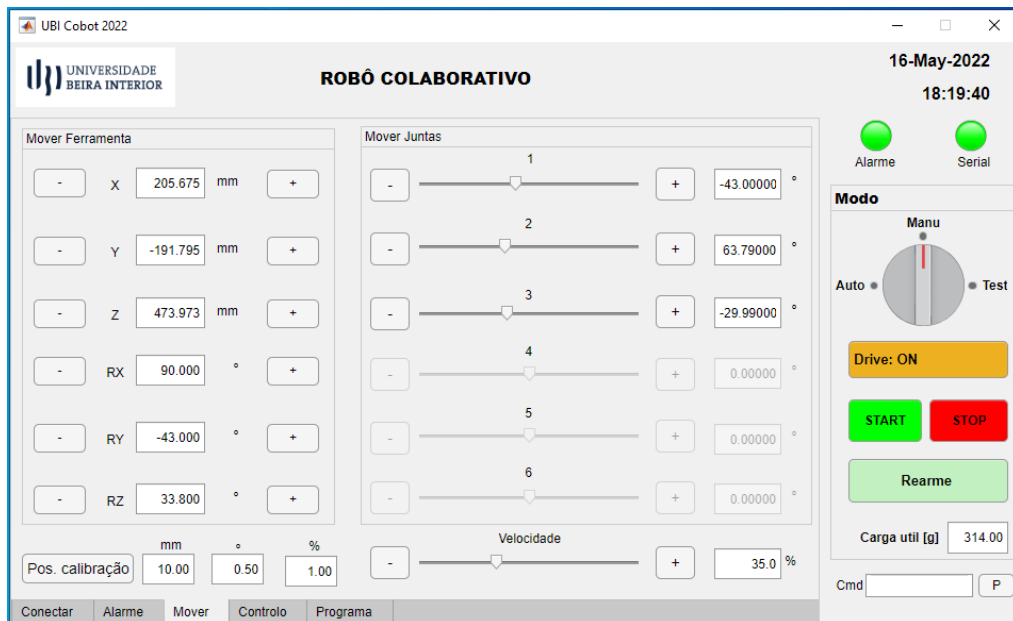


b) Teste n.º 3, posição, orientação e a ângulo das juntas do robô na posição P2.

Fig. 49. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).

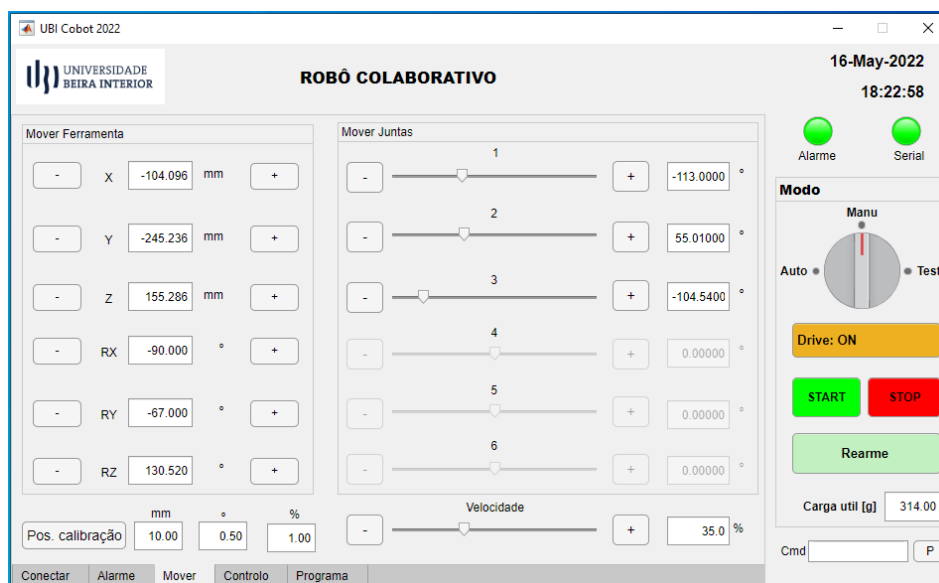


c) Teste n.º 3, posição, orientação e a ângulo das juntas do robô na posição P3.



d) Teste n.º 3, posição, orientação e a ângulo das juntas do robô na posição P4.

Fig. 50. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).



e) Teste n.º 3, posição, orientação e a ângulo das juntas do robô na posição P5.



f) Teste n.º 3, robô na posição P4.

Fig. 51. Robô em diferentes posições durante o teste n.º 3 de precisão e repetibilidade (cont.).

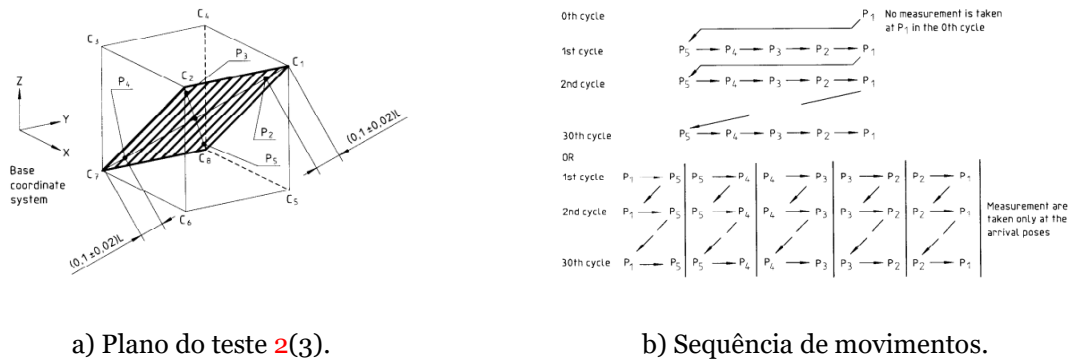


Fig. 52. Área de trabalho com posições e sequência de pontos para o teste de precisão e de repetibilidade.

A Tabela 19 apresenta as coordenadas das 5 posições (P1, ... P5) que constam no plano da Figura 47.

Tabela 20. Ângulos, posições e orientações escolhidas para teste de repetibilidade.

Posição	Ângulos das juntas [°]			Posição [mm]			Orientação [°]		
	θ_1	θ_2	θ_3	X	Y	Z	X	Y	Z
P1	-81,0	54,9	-92,5	46,8	-295,8	189,57	90,0	-81,0	-37,6
P2	-100,0	10,0	0,0	-74,2	-420,9	238,4	-90,0	-80,0	-170,0
P3	-61,0	25,0	-30,0	200,7	-362,1	229,2	90,0	-61,0	-5,0
P4	-43,0	63,8	-30,0	205,6	-191,7	473,9	90,0	-43,0	33,8
P5	-113,0	55,0	-104,5	-104,1	-245,2	155,2	-90,0	-67,0	130,5

A Tabela 20 apresenta o ângulo de cada junta para cada posição relacionado com cada ciclo. A Tabela 21 apresenta a posição e na Tabela 22 encontra-se a orientação. Na Tabela 20, os 15 primeiros valores de ângulo de cada junta foram obtidos com 10% de velocidade e os últimos 15 valores com 35% de velocidade. A partir da Tabela 21, calcula-se a precisão e a repetibilidade (ISO, 1998) como:

1) Precisão de posição (AP_P) como:

$$AP_P = \sqrt{AP_x^2 + AP_y^2 + AP_z^2} \quad (\text{eq. 19})$$

$$AP_x = (\bar{x} - x_c) \quad (\text{eq. 19. a})$$

$$AP_y = (\bar{y} - y_c) \quad (\text{eq. 19. b})$$

$$AP_z = (\bar{z} - z_c) \quad (\text{eq. 19. c})$$

Onde \bar{x} , \bar{y} e \bar{z} são as médias das posições obtidas e x_c , y_c e z_c são as posições comandadas.

A repetibilidade de posicionamento (RP_l) é calculada como:

$$RP_l = \bar{l} + 3 \cdot S_l \quad (\text{eq. 19. d})$$

$$\bar{l} = \frac{1}{n} \cdot \sum_{j=1}^n l_j \quad (\text{eq. 19. e})$$

$$l_j = \sqrt{(\bar{x} - x_j)^2 + (\bar{y} - y_j)^2 + (\bar{z} - z_j)^2} \quad (\text{eq. 19. f})$$

Onde x_j , y_j e z_j são os ângulos da pose j-th alcançada

$$S_l = \sqrt{\frac{\sum_{j=1}^n (l_j - \bar{l})^2}{n - 1}} \quad (\text{eq. 19. g})$$

Similarmente a precisão e a repetibilidade de orientação (o) é dada por:

$$AP_o = \sqrt{(\bar{O} - O_c)^2} \quad (\text{eq. 19. i})$$

$$RP_b = \pm 3 \cdot S_o = \pm 3 \cdot \sqrt{\frac{\sum_{j=1}^n (O_j - \bar{O})^2}{n - 1}} \quad (\text{eq. 19. j})$$

Os resultados da precisão e a repetibilidade calculado encontra-se compilado na Tabela 23.

Tabela 21. Ângulos de cada junta registrada durante o teste de repetibilidade e precisão.

Ciclo	P1			P2			P3			P4			P5		
	θ_1 [°]	θ_2 [°]	θ_3 [°]	θ_1 [°]	θ_2 [°]	θ_3 [°]	θ_1 [°]	θ_2 [°]	θ_3 [°]	θ_1 [°]	θ_2 [°]	θ_3 [°]	θ_1 [°]	θ_2 [°]	θ_3 [°]
1	-81,00	54,90	-92,53	-100,00	10,01	0,01	-61,00	25,00	-30,03	-43,00	63,79	-29,99	-113,00	55,01	-104,53
2	-81,00	54,91	-92,46	-100,00	10,01	0,02	-61,00	25,00	-30,03	-43,00	63,79	-29,99	-113,00	55,00	-104,52
3	-81,00	54,91	-92,48	-100,00	10,01	0,01	-61,00	24,99	-30,01	-43,00	63,79	-30,03	-113,00	55,01	-104,50
4	-81,00	54,91	-92,48	-100,00	10,01	0,01	-61,00	24,99	-30,03	-43,00	63,79	-29,99	-112,99	55,01	-104,52
5	-81,00	54,91	-92,48	-99,99	10,01	0,01	-61,01	24,99	-30,03	-43,00	63,79	-29,99	-113,00	55,01	-104,52
6	-81,01	54,91	-92,48	-100,00	10,01	0,01	-61,01	24,99	-30,02	-43,00	63,79	-29,99	-113,00	55,01	-104,52
7	-81,01	54,91	-92,47	-100,00	10,01	0,01	-61,01	24,99	-30,03	-43,00	63,79	-29,99	-112,99	55,01	-104,52
8	-81,01	54,91	-92,48	-100,00	10,01	0,01	-61,01	24,99	-30,03	-43,00	63,79	-29,98	-113,00	55,01	-104,50
9	-81,01	54,91	-92,49	-100,00	10,01	0,02	-61,00	25,00	-30,02	-43,00	63,79	-29,99	-113,00	55,01	-104,52
10	-81,01	54,91	-92,48	-100,00	10,01	0,02	-61,01	25,00	-30,02	-43,00	63,79	-29,99	-112,99	55,01	-104,50
11	-81,01	54,91	-92,49	-99,99	10,01	0,02	-61,01	25,00	-30,03	-43,00	63,79	-29,99	-113,00	55,01	-104,53
12	-81,00	54,91	-92,47	-99,99	10,01	0,01	-61,01	24,99	-30,02	-43,00	63,79	-29,98	-113,00	55,01	-104,52
13	-81,01	54,91	-92,48	-100,00	10,01	0,01	-61,01	24,99	-30,02	-43,00	63,79	-29,99	-113,00	55,01	-104,52
14	-81,00	54,91	-92,48	-100,00	10,01	0,02	-61,01	24,99	-30,01	-43,00	63,80	-30,03	-113,00	55,01	-104,50
15	-81,01	54,91	-92,48	-100,00	10,01	0,02	-61,01	25,00	-30,03	-43,00	63,79	-29,99	-113,00	55,01	-104,52
16	-81,00	54,91	-92,52	-100,00	10,01	0,07	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-112,99	55,00	-104,54
17	-81,01	54,91	-92,46	-100,00	10,01	0,04	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
18	-81,00	54,91	-92,46	-99,99	10,01	0,04	-61,00	25,01	-29,99	-43,00	63,79	-29,99	-113,00	55,01	-104,57
19	-81,00	54,91	-92,52	-100,00	10,00	0,07	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-112,99	55,00	-104,54
20	-81,01	54,91	-92,46	-100,00	10,01	0,04	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
21	-81,00	54,91	-92,46	-99,99	10,01	0,04	-61,00	25,01	-29,99	-43,00	63,79	-29,99	-113,00	55,01	-104,57
22	-81,00	54,91	-92,47	-99,99	10,01	0,02	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
23	-81,00	54,91	-92,47	-100,00	10,01	0,02	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-112,99	55,01	-104,52
24	-81,00	54,91	-92,47	-100,00	9,99	0,02	-61,00	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
25	-81,01	54,91	-92,47	-100,00	10,01	0,02	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
26	-81,00	54,91	-92,47	-100,00	10,01	0,04	-61,00	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,54
27	-81,00	54,91	-92,46	-100,00	10,01	0,04	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
28	-81,00	54,91	-92,50	-100,00	10,00	0,07	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,54
29	-81,01	54,91	-92,46	-100,00	10,01	0,01	-61,00	25,01	-29,98	-43,00	63,79	-29,98	-113,00	55,01	-104,57
30	-81,00	54,91	-92,47	-99,99	10,01	0,02	-61,01	25,01	-30,03	-43,00	63,79	-30,03	-113,00	55,01	-104,53
Média	-81,004	54,910	-92,478	-99,998	10,009	0,026	-61,007	25,002	-30,023	-43,000	63,790	-30,008	-112,998	55,009	-104,528
Desvio	0,005	0,002	0,018	0,004	0,004	0,018	0,005	0,009	0,013	0,000	0,002	0,021	0,004	0,003	0,018

Tabela 22. Posição do efector registada durante o teste de repetibilidade e precisão.

Ciclo	P1			P2			P3			P4			P5		
	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]
1	46,845	-295,767	189,474	-74,215	-420,893	238,477	200,712	-362,093	229,048	205,675	-191,795	473,973	-104,049	-245,123	155,182
2	46,871	-295,932	189,749	-74,214	-420,887	238,517	200,712	-362,093	229,048	205,675	-191,795	473,973	-104,060	-245,150	155,162
3	46,863	-295,884	189,685	-74,215	-420,893	238,477	200,721	-362,110	229,056	205,741	-191,856	473,840	-104,085	-245,208	155,260
4	46,863	-295,884	189,685	-74,215	-420,893	238,477	200,717	-362,104	228,976	205,675	-191,795	473,973	-104,018	-245,170	155,208
5	46,863	-295,884	189,685	-74,141	-420,906	238,477	200,654	-362,139	228,976	205,675	-191,795	473,973	-104,061	-245,152	155,208
6	46,812	-295,892	189,685	-74,215	-420,893	238,477	200,656	-362,142	229,016	205,675	-191,795	473,973	-104,061	-245,152	155,208
7	46,816	-295,916	189,717	-74,215	-420,893	238,477	200,654	-362,139	228,976	205,675	-191,795	473,973	-104,018	-245,170	155,208
8	46,812	-295,892	189,685	-74,215	-420,893	238,477	200,654	-362,139	228,976	205,659	-191,780	474,006	-104,085	-245,208	155,260
9	46,808	-295,868	189,653	-74,214	-420,887	238,517	200,713	-362,097	229,088	205,675	-191,795	473,973	-104,061	-245,152	155,208
10	46,812	-295,892	189,685	-74,214	-420,887	238,517	200,650	-362,132	229,088	205,675	-191,795	473,973	-104,042	-245,226	155,260
11	46,808	-295,868	189,653	-74,140	-420,900	238,517	200,648	-362,128	229,048	205,675	-191,795	473,973	-104,049	-245,123	155,182
12	46,867	-295,908	189,717	-74,141	-420,906	238,477	200,656	-362,142	229,016	205,659	-191,780	474,006	-104,061	-245,152	155,208
13	46,812	-295,892	189,685	-74,215	-420,893	238,477	200,656	-362,142	229,016	205,675	-191,795	473,973	-104,061	-245,152	155,208
14	46,863	-295,884	189,685	-74,214	-420,887	238,517	200,657	-362,145	229,056	205,701	-191,819	473,889	-104,085	-245,208	155,260
15	46,812	-295,892	189,685	-74,214	-420,887	238,517	200,648	-362,128	229,048	205,675	-191,795	473,973	-104,061	-245,152	155,208
16	46,848	-295,787	189,558	-74,208	-420,852	238,715	200,643	-362,118	229,121	205,741	-191,856	473,840	-103,993	-245,112	155,109
17	46,819	-295,940	189,749	-74,211	-420,873	238,596	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
18	46,871	-295,932	189,749	-74,138	-420,886	238,596	200,713	-362,096	229,281	205,675	-191,795	473,973	-104,001	-245,011	155,078
19	46,848	-295,787	189,558	-74,210	-420,865	238,640	200,643	-362,118	229,121	205,741	-191,856	473,840	-103,993	-245,112	155,109
20	46,819	-295,940	189,749	-74,211	-420,873	238,596	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
21	46,871	-295,932	189,749	-74,138	-420,886	238,596	200,713	-362,096	229,281	205,675	-191,795	473,973	-104,001	-245,011	155,078
22	46,867	-295,908	189,717	-74,140	-420,900	238,517	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
23	46,867	-295,908	189,717	-74,214	-420,887	238,517	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,018	-245,170	155,208
24	46,867	-295,908	189,717	-74,218	-420,913	238,368	200,706	-362,083	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
25	46,816	-295,916	189,717	-74,214	-420,887	238,517	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
26	46,867	-295,908	189,717	-74,211	-420,873	238,596	200,706	-362,083	229,121	205,741	-191,856	473,840	-104,037	-245,095	155,156
27	46,871	-295,932	189,749	-74,211	-420,873	238,596	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
28	46,856	-295,835	189,621	-74,210	-420,865	238,640	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,037	-245,095	155,156
29	46,819	-295,940	189,749	-74,215	-420,893	238,477	200,715	-362,099	229,321	205,659	-191,780	474,006	-104,001	-245,011	155,078
30	46,867	-295,908	189,717	-74,140	-420,900	238,517	200,643	-362,118	229,121	205,741	-191,856	473,840	-104,049	-245,123	155,182
Media	46,843	-295,891	189,689	-74,196	-420,887	238,529	200,670	-362,117	229,092	205,703	-191,821	473,916	-104,043	-245,133	155,183
Desvio	0,025	0,044	0,063	0,031	0,013	0,068	0,031	0,018	0,085	0,034	0,031	0,069	0,026	0,052	0,050

Tabela 23. Orientação do efector registada durante o teste de repetibilidade e precisão.

Ciclo	P1			P2			P3			P4			P5		
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
1	90,00	-81,00	-37,63	-90,00	-80,00	-169,98	90,00	-61,00	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,48
2	90,00	-81,00	-37,55	-90,00	-80,00	-169,97	90,00	-61,00	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,48
3	90,00	-81,00	-37,57	-90,00	-80,00	-169,98	90,00	-61,00	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,51
4	90,00	-81,00	-37,57	-90,00	-80,00	-169,98	90,00	-61,00	-5,04	90,00	-43,00	33,80	-90,00	-67,01	130,49
5	90,00	-81,00	-37,57	-90,00	-80,01	-169,98	90,00	-61,01	-5,04	90,00	-43,00	33,80	-90,00	-67,00	130,49
6	90,00	-81,01	-37,57	-90,00	-80,00	-169,98	90,00	-61,01	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,49
7	90,00	-81,01	-37,56	-90,00	-80,00	-169,98	90,00	-61,01	-5,04	90,00	-43,00	33,80	-90,00	-67,01	130,49
8	90,00	-81,01	-37,57	-90,00	-80,00	-169,98	90,00	-61,01	-5,04	90,00	-43,00	33,81	-90,00	-67,00	130,51
9	90,00	-81,01	-37,58	-90,00	-80,00	-169,97	90,00	-61,00	-5,02	90,00	-43,00	33,80	-90,00	-67,00	130,49
10	90,00	-81,01	-37,57	-90,00	-80,00	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,80	-90,00	-67,01	130,51
11	90,00	-81,01	-37,58	-90,00	-80,01	-169,97	90,00	-61,01	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,48
12	90,00	-81,00	-37,56	-90,00	-80,01	-169,98	90,00	-61,01	-5,03	90,00	-43,00	33,81	-90,00	-67,00	130,49
13	90,00	-81,01	-37,57	-90,00	-80,00	-169,98	90,00	-61,01	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,49
14	90,00	-81,00	-37,57	-90,00	-80,00	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,77	-90,00	-67,00	130,51
15	90,00	-81,01	-37,57	-90,00	-80,00	-169,97	90,00	-61,01	-5,03	90,00	-43,00	33,80	-90,00	-67,00	130,49
16	90,00	-81,00	-37,61	-90,00	-80,00	-169,92	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,01	130,46
17	90,00	-81,01	-37,55	-90,00	-80,00	-169,95	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
18	90,00	-81,00	-37,55	-90,00	-80,01	-169,95	90,00	-61,00	-4,98	90,00	-43,00	33,80	-90,00	-67,00	130,44
19	90,00	-81,00	-37,61	-90,00	-80,00	-169,93	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,01	130,46
20	90,00	-81,01	-37,55	-90,00	-80,00	-169,95	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
21	90,00	-81,00	-37,55	-90,00	-80,01	-169,95	90,00	-61,00	-4,98	90,00	-43,00	33,80	-90,00	-67,00	130,44
22	90,00	-81,00	-37,56	-90,00	-80,01	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
23	90,00	-81,00	-37,56	-90,00	-80,00	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,01	130,49
24	90,00	-81,00	-37,56	-90,00	-80,00	-169,99	90,00	-61,00	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
25	90,00	-81,01	-37,56	-90,00	-80,00	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
26	90,00	-81,00	-37,56	-90,00	-80,00	-169,95	90,00	-61,00	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,47
27	90,00	-81,00	-37,55	-90,00	-80,00	-169,95	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
28	90,00	-81,00	-37,59	-90,00	-80,00	-169,93	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,47
29	90,00	-81,01	-37,55	-90,00	-80,00	-169,98	90,00	-61,00	-4,97	90,00	-43,00	33,81	-90,00	-67,00	130,44
30	90,00	-81,00	-37,56	-90,00	-80,01	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,76	-90,00	-67,00	130,48
Média	90,00	-81,00	-37,57	-90,00	-80,00	-169,97	90,00	-61,01	-5,02	90,00	-43,00	33,78	-90,00	-67,00	130,48
Desvio	0,00	0,00	0,02	0,00	0,00	0,02	0,00	0,00	0,02	0,00	0,00	0,02	0,00	0,00	0,02

Tabela 24. Precisão e repetibilidade obtida durante os testes.

Posição	Precisão [mm]	Repetibilidade [mm]	Precisão [°]	Repetibilidade [°]
P5	APp = 0,090 APx = 0,041 APy = 0,074 APz = -0,031	RP = 0,2811	APo = 0,0191	RPo = ± 0,0459
P4	APp = 0,100 APx = 0,051 APy = -0,047 APz = -0,072	RP = 0,3423	APo = 0,0173	RPo = ± 0,0579
P3	APp = 0,091 APx = -0,047 APy = -0,014 APz = -0,076	RP = 0,254	APo = 0,0217	RPo = ± 0,0213
P2	APp = 0,170 APx = 0,022 APy = 0,026 APz = 0,166	RP = 0,3835	APo = 0,0344	RPo = ± 0,0524
P1	APp = 0,131 APx = -0,013 APy = -0,051 APz = 0,119	RP = 0,2836	APo = 0,0316	RPo = ± 0,0364

Os resultados do teste n.º 3 de precisão e repetibilidade vêm da comparação da Tabela 20 com os dados da Tabela 19. É apresentado um desvio máximo de 0,021° nos ângulos de cada juntas em relação à média. É apresentada uma variação máxima até 0,07° nos ângulos de cada juntas durante os ciclos. Estes resultados são mais afetados nos ciclos em que a percentagem da velocidade é de 35% (os últimos 15 ciclos). Comparando com os resultados expostos na Tabela 21, é apresentado um desvio máximo de 0,085 mm em relação à média das posições finais do efector, e uma variação de 0,179 mm durante os ciclos do movimento em ambas percentagens de velocidade (10% e 35%). Esta variação foi máxima ao atingir a posição P1. Por fim, comparando com os resultados expostos na Tabela 22, apresenta-se um desvio máximo de 0,02° em relação à média das orientações do efector, e uma variação de 0,08° durante o movimento nos ciclos. Pode-se verificar que uma pequena variação nos ângulos das junto origina uma variação significativa na posição final do efector.

5.4. Teste N.º 4: Verificação da cinemática inversa

O teste n.º 4 avalia a eficiência do cálculo da cinemática inversa. Para tal pretendeu-se alcançar as posições e orientações da Tabela 17 sempre partindo da posição de origem. Para efetuar o teste definiram-se 150 partículas e para critério de paragem, o número máximo de iterações 1000, um erro de 0,1% de erro inicial e tempo máximo processamento de 10 segundos (Alkayyali & Tutunji, 2019). A Tabela 24 apresenta os resultados obtidos no teste da eficiência no cálculo da cinemática inversa.

Tabela 25. Resultado da eficiência no cálculo da cinemática inversa.

Posição	Ângulo das juntas			Posição [mm]		
	θ_1	θ_2	θ_3	X	Y	Z
P1	-81,44	58,43	-96,59	42,80	-284,45	194,75
P2	-100,70	9,85	1,43	-79,23	-419,12	242,92
P3	-61,54	18,63	-17,27	201,70	-372,10	233,63
P4	-44,06	63,05	-123,63	147,63	-142,86	144,51
P5	-115,25	57,09	-105,81	-112,00	-237,49	161,42
	Orientação [°]			Iteração	% de erro	Tempo [ms]
Posição	X	Y	Z			
P1	90	-81,44	-38,15	10	0,091	0,2
P2	-90	-79,3	-168,7	7	0,012	0,127
P3	90	-61,54	1,36	10	0,073	0,256
P4	90	-44,06	-60,58	10	0,068	0,249
P5	-90	-64,75	131,28	11	0,065	0,311

Os resultados do teste n.º 4, apresentado na Tabela 24 e quando comparado com os resultados da Tabela 16, apresentam uma grande eficiência na convergência do algoritmo PSO em pouco tempo de processamento (no máximo 0,13 segundos) e no máximo 11 iteração. Embora a posição e orientação apresente um erro máximo de 10,7 mm e 4,7°, estes desvios devem-se ao facto de que o erro mínimo pretendido é de 0,1%. A comparação com o ângulo inicial da junta com o ângulo final estimado pelo algoritmo é indiferente porque na cinemática inversa existem várias soluções para o mesmo problema.

5.5. Teste N.º 5: Corrente eléctrica vs. Carga no efector

O teste n.º 5 consistiu na verificação de comportamento da corrente eléctrica com a variação de carga (peso) no efector. Foram seleccionadas 4 posições diferentes e a trajetória do braço foi sempre considerada partindo da posição de origem P (0,0,0), sendo (0,0,0) correspondente ao ângulo [°] das 3 juntas e P (-50,50,50) [0x] corresponde à corrente no eixo da junta x influenciada

pelo movimento nos 3 eixos. As 4 posições são: 1 - P (-50,0,0), apenas variação no eixo da junta 1; 2 - P (0,50,0), apenas variação no eixo da junta 2; 3- P (0,0,50), apenas variação no eixo da junta 3; e 4 - P (-50,50,50), variação nos 3 eixos. A Figura 51 representa o resultado da corrente em função de peso obtido durante teste n° 5.

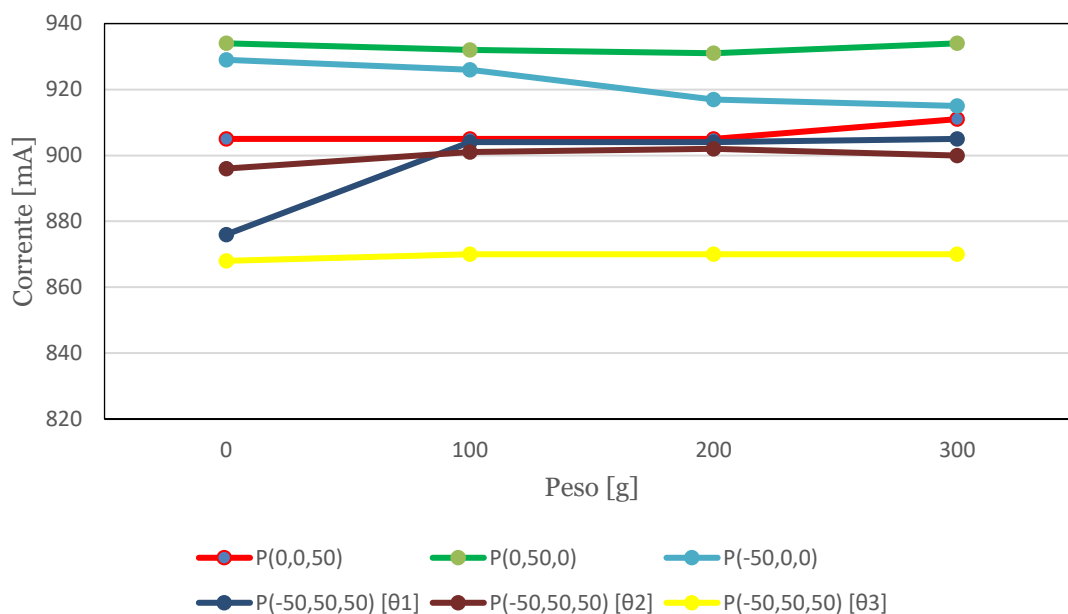


Fig. 53. Resultado da corrente em função de peso obtido durante teste n.º 5.

Pela análise da Figura 51, conclui-se que:

- Para peso inferior a 100 g, a corrente elétrica é baixa, o que faz com que a corrente medida seja praticamente influenciada pelo ganho de modo comum do amplificador operacional conforme Equação 16.h.

- É possível distinguir o aumento da corrente com aumento de peso (P (0,50,0) e P (0,0,50)), quando apenas for o movimento em apenas 1 eixo de cada junta. Não foi possível verificar o mesmo para P (-50,0,0), pois a variação da corrente foi reduzida, sendo influenciada pelo ganho diferencial.

- Não é significativo o aumento da corrente com aumento de peso durante o movimento de P (0,0,0) para P (-50,50,50), pois trata-se de um movimento em mais de uma junta, uma vez que a força exercida depende de ângulo das outras juntas que estão a variar. Isso faz com que o gráfico P (0,50,0) seja diferente de P (0,50,0) [θ2], P (0,0,50) diferente de P (0,0,50) [θ3].

5.6. Teste N.º 6: Detecção de colisões

O teste n.º 6 residiu na análise da técnica e método para detetar colisões. Para efetuar este teste foram percorridas 5 posições descritas na Tabela 16 no modo automático em que a rede neuronal artificial treinada estima a corrente máxima atingida em cada eixo com base no erro, peso acoplado no efector e a percentagem de velocidade de execução. A rede foi treinada com 290 dados obtidos com robô em modo execução teste. A Figura 52 representa o gráfico com a corrente máxima estimada pela rede e medida durante execução.

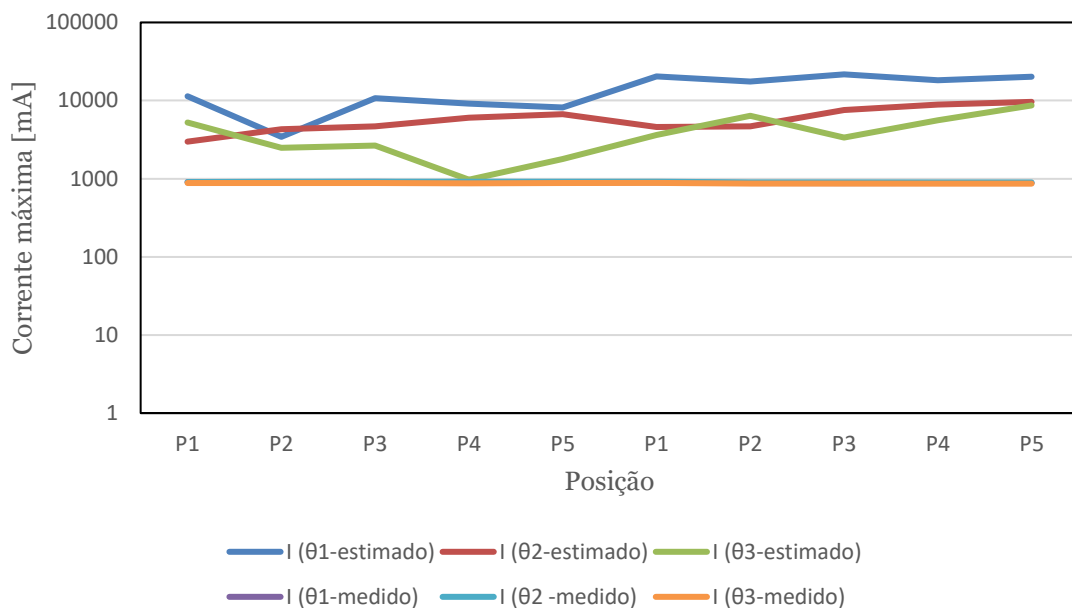


Fig. 54. Resultado da corrente máxima estimada e medida durante o teste n.º 6.

As 5 primeiras posições (P1 ... P5) durante o teste foram alcançadas com a velocidade de 20% e com um peso de 200g, e as 5 últimas foram alcançadas com a velocidade a 35% e 300 g de peso.

Durante o teste não foi detetada a colisão, como seria de esperar, mas graficamente observa-se que a corrente estimada pela rede aumenta com aumento de peso e velocidade. Em contrapartida, estes valores são muito elevados em relação aos picos de corrente atingidos durante a execução, o que faz com que o robô nunca detetasse a colisão mesmo se colidisse. A corrente medida durante o teste sofreu poucas oscilações devido à velocidade e peso, mas inclui um erro associado ao ganho de modo comum da montagem diferencial.

Dado o resultado pouco satisfatório obtido durante o teste e para que o braço robótico ficasse com uma proteção caso colidisse, recorreu-se aos 290 dados obtidos e calculou-se a corrente máxima

obtida em cada eixo acrescentando mais 2 % e definiu como a corrente máxima de segurança para colisão. A corrente máxima de segurança para colisão ficou definida em 946 mA para eixo 1, 948 mA para eixo 2 e 915 mA para eixo 3.

5.7. Nota conclusiva

Ao longo deste capítulo foram descritas todas as etapas para a obtenção de alguns parâmetros e características do manipulado robótico desenvolvido. Durante a montagem, ao efetuar o teste n.º 1 não foi possível a montagem total do manipulador devido ao peso, sendo que os testes foram apenas efetuados com os 3 primeiros eixos, com 10% ou 30% da velocidade e ambos em 1/8 de passo.

6. Conclusões

6.1. Conclusões gerais

O desenvolvimento do robô colaborativo de baixo custo foi um projeto bastante complexo de desenvolver devido à forte componente prática, controle e programação, tendo-se revelado um desafio enorme abraçar a sua construção. Devido à quantidade de trabalho, o mesmo tornou-se ambicioso, uma vez que se trata de uma estrutura impressa na impressora 3D, á base de plástico, desde a estrutura, caixa de engrenagem até ao pinhão mecânico. Ao longo do desenvolvimento do robô, foram aparecendo alguns problemas construtivos como espaço para conter a eletrónica, espaço e caminho de modo a manter a fiação oculta não visível.

Quanto ao controle no posicionamento, foram desenvolvidos vários testes em que se denota grande desempenho do motor de passo e método de controle proposto, apresentando um erro máximo no posicionamento de $0,035 [^\circ]$, quando o bloqueio do eixo do motor é assegurado pelo driver. Todavia, origina o aquecimento do motor o que leva à dilatação do pinhão que está acoplado ao eixo do motor causando o desacoplamento mecânico. Quando o bloqueio de movimento é assegurado pelo binário criado pela caixa de engrenagem (driver desabilitada), a variação é maior devido à folga, desgaste da coroa dentada e o pinhão. Estas variações em ambos os casos tende a aumentar com o aumento de peso acoplado ao braço robótico.

Quanto ao método e algoritmo aplicado, a cinemática inversa, houve sempre a convergência na solução o que demonstra grande eficiência do algoritmo PSO, mas, entretanto, este tem erro associado e tempo de processamento dependendo dos parâmetros pré-definidos.

Para a maioria dos atuais manipuladores industriais, a repetibilidade é muito alta, atingindo valores abaixo de $0,1 \text{ mm}$, enquanto a precisão média está em torno de 5 e 15 mm (Abderrahim *et al.*, 2006). Através da Tabela 23, pode-se observar que houve um bom desempenho e resultado a nível de precisão, sendo o erro máximo de $0,17 \text{ mm}$ no P2. Já o erro de repetibilidade apresentou valores aceitável, tendo um valor máximo em P2 com $RP = 0,384 \text{ mm}$.

O método e estratégia para a proteção e deteção de colisão continua a ser um desafio na robótica, principalmente nos robôs colaborativos. O método proposto continua a ser um desafio sendo que

requer muitos dados de teste com menos erro associado, propondo-se a otimização no circuito para a medição de corrente elétrica.

Dos testes que foram realizados aos 3 eixos do ao manipulador robótico, os ângulos alcançáveis para cada eixo são: eixo 1, -180 a 180 [°], eixo 2, -30 a 210 [°] e eixo 3, -160 a 160 [°]. O braço pode alcançar até 434 mm na horizontal e uma altura até 597 mm.

De um modo geral, a construção de um robô colaborativo foi um grande desafio, por unificar as três áreas de domínio de um engenheiro mecatrónico, a elétrica, a mecânica e a computação. O conceito colaborativo continuará a ser desafiador o que leva a ter custos elevado.

6.2. Sugestões de trabalhos futuros

Devido à abrangência deste projeto ficaram inúmeras ideias por concretizar e, por isso, serão enumeradas de seguida algumas propostas para trabalhos futuros.

Em primeiro lugar, sugere-se algumas alterações na estrutura no que diz respeito ao especto construtivo do braço, para conter diretamente os componentes eletrónicos, reforçar a estrutura para suportar mais peso, aumentar a potência dos motores e redimensionar a caixa redutora para aumentar o binário.

De seguida, seria interessante o estudo mais profundo da mecânica do braço robótico, nomeadamente a estática (força e binário) e a dinâmica para auxiliar no processo de controlo e também o estudo mais complexo no que diz respeito à cinemática inversa, com o intuito de reduzir o erro.

Por fim, programar e aplicar outros tipos de movimentos sem ser das juntas, como sejam movimentos lineares ou circulares, conforme existe na literatura aplicados nos robôs indústrias e entre outros. Também desenvolver uma aplicação para interface do utilizador de modo a possibilitar o movimento e controlo do braço robótico a partir de aparelhos móveis (tablet, telemóveis, iPad, etc.), ou mesmo projetar e desenvolver uma consola ligada diretamente ao braço robótico conforme existe nos robôs comercializáveis.

Referências Bibliográficas

- 3D Lab. (n.d.). *3D Lab*. Retrieved October 19, 2021, from <https://3dlab.com.br/10-sofwares-de-modelagem-3d/>
- A. Gasparetto, & Scalerà, L. (2019). A Brief History of Industrial Robotics in the 20th Century. *Advances in Historical Studies Publication*. <https://www.scirp.org/journal/paperinformation.aspx?paperid=90517>
- ABB. (n.d.). *Technical data IRB 14000 YuMi*. Retrieved November 25, 2020, from <https://new.abb.com/products/robotics/collaborative-robots/irb-14000-yumi/irb-14000-yumi-data>
- Abderrahim, M., Khamis, A., Garrido, S., & Moreno, L. (2006). Accuracy and Calibration Issues of Industrial Manipulators. *Industrial Robotics: Programming, Simulation and Applications, June 2014*. <https://doi.org/10.5772/4895>
- Alkayyali, M., & Tutunji, T. A. (2019). PSO-based algorithm for inverse kinematics solution of robotic arm manipulators. *Proceedings of the 2019 20th International Conference on Research and Education in Mechatronics, REM 2019, 5, 8–13*. <https://doi.org/10.1109/REM.2019.8744103>
- All3DP. (2021). *All3DP*. <https://all3dp.com/2/pla-vs-abs-filament-3d-printing/>
- Arduino. (n.d.). *No Title*. Retrieved October 18, 2021, from <https://www.arduino.cc/en/Guide/Environment>
- Arm, T. F. S. R., Liang, X., Cheong, H., Sun, Y., Member, S., & Guo, J. (2018). *Design , Characterization , and Implementation of a. 3(3), 2702–2709*.
- Arrow.com. (2020). *No Title*. <https://www.arrow.com/en/research-and-events/articles/engineering-basics-what-is-a-microcontroller>
- Babiuch, M., Foltynnek, P., & Smutny, P. (2019). Using the ESP32 microcontroller for data processing. *Proceedings of the 2019 20th International Carpathian Control Conference, ICC 2019*. <https://doi.org/10.1109/CarpathianCC.2019.8765944>
- Bagdalkar, P., & Ali, L. (2020). Interfacing of light sensor with FPGA using I2C bus. *2020 6th*

- International Conference on Advanced Computing and Communication Systems, ICACCS 2020*, 843–846. <https://doi.org/10.1109/ICACCS48705.2020.9074372>
- Bipasha Biswas, S., & Tariq Iqbal, M. (2018). Solar Water Pumping System Control Using a Low Cost ESP32 Microcontroller. *Canadian Conference on Electrical and Computer Engineering, 2018-May*, 7–11. <https://doi.org/10.1109/CCECE.2018.8447749>
- Bolla, D. R., Jijesh, J. J., Palle, S. S., Penna, M., Keshavamurthy, & Shivashankar. (2020). An IoT Based Smart E-Fuel Stations Using ESP-32. *Proceedings - 5th IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2020*, 333–336. <https://doi.org/10.1109/RTEICT49044.2020.9315676>
- Bunkum, M., Vachirasakulchai, P., Nampeng, J., Tommajaree, R., & Visitsattapongse, S. (2019). Tele-Operation of Robotic Arm. *BMEiCON 2019 - 12th Biomedical Engineering International Conference*. <https://doi.org/10.1109/BMEiCON47515.2019.8990247>
- Colgate, J. E., & Peshkin, M. A. (1999). *United States Patent US5952796A - Cobots*. 19. <https://patents.google.com/patent/US5952796A/en?q=cobot&oq=cobot>
- Correia Simões, A., Lucas Soares, A., & Barros, A. C. (2020). Factors influencing the intention of managers to adopt collaborative robots (cobots) in manufacturing organizations. *Journal of Engineering and Technology Management - JET-M*, 57(May), 101574. <https://doi.org/10.1016/j.jengtecman.2020.101574>
- Dasari, A., & Reddy, N. S. (2012). Forward and inverse kinematics of a robotic frog. *4th International Conference on Intelligent Human Computer Interaction: Advancing Technology for Humanity, IHCI 2012*. <https://doi.org/10.1109/IHCI.2012.6481850>
- Dizon, J. R. C., Espera, A. H., Chen, Q., & Advincula, R. C. (2018). Mechanical characterization of 3D-printed polymers. *Additive Manufacturing*, 20, 44–67. <https://doi.org/10.1016/j.addma.2017.12.002>
- Electronics Hub. (2019). *No Title*. <https://www.electronicshub.org/position-sensors/>
- ElectroPark. (n.d.). *ElectroPark.pl, Drukarka 3D K8200 - KIT*. Retrieved December 15, 2020, from <http://electropark.pl/drukarki-3d/4918-drukarka-3d-k8200-kit.html>
- ES927 – Controlo de Sistemas Robóticos Cinemática e Dinâmica de Robôs Nota ao leitor Este material é baseado principalmente na referência : (2015).*
- Espressif. (n.d.). *No Title*. Retrieved August 1, 2021, from <https://www.espressif.com/en/products/socs/esp32>

- Fang, Y. Y., & Chen, X. J. (2011). Design and simulation of UART serial communication module based on VHDL. *2011 3rd International Workshop on Intelligent Systems and Applications, ISA 2011 - Proceedings*. <https://doi.org/10.1109/ISA.2011.5873448>
- FANUC. (n.d.). *Collaborative Robot CR-4iA*. Retrieved November 30, 2020, from <https://www.fanuc.eu/pt/pt/robôs/página-filtro-robôs/robôs-colaborativos/collaborative-cr4ia>
- Fernández, E. F., Almonacid, F., Sarmah, N., Rodrigo, P., Mallick, T. K., & Pérez-Higueras, P. (2014). A model based on artificial neuronal network for the prediction of the maximum power of a low concentration photovoltaic module for building integration. *Solar Energy*, *100*, 148–158. <https://doi.org/10.1016/j.solener.2013.11.036>
- Gnd_To_Vcc. (n.d.). *No Title*. Retrieved October 5, 2021, from <https://gndtovcc.home.blog/2020/04/12/lets-start-with-esp-now-esp8266-nodemcu-with-arduino-ide/>
- Hubs. (n.d.). *No Title*. Retrieved July 23, 2021, from <https://www.hubs.com/knowledge-base/pla-vs-abs-whats-difference/#heat-resistance>
- IFR. (2016). *Industrial robots - definition and classification Industrial*. 29–42. https://ifr.org/img/office/Industrial_Robots_2016_Chapter_1_2.pdf
- IFR. (2020). World Robotics 2020 Industrial Robots; World Robotics 2020 Service Robots. <https://ifr.org/>
- International Organisation of Standardization. (1998). *INTERNATIONAL STANDARD ISO 9283 Second edition*. 60. http://106.38.59.21:8080/userfiles/d894aa13635f4f70ae241466d6504b6e/files/teckSolution/2019/12/ISO_9283-1998.pdf
- ISO 15066. (2016). <https://www.iso.org/obp/ui/#iso:std:iso:ts:15066:ed-1:v1:en>
- J. Norberto Pires. (2018). *Robótica Industrial* (LIDEL (Ed.); 1st ed.). <https://www.jnorbertopires.pt>
- Karmoker, S., Polash, M. M. H., & Hossan, K. M. Z. (2014). Design of a low cost PC interface Six DOF robotic arm utilizing recycled materials. *1st International Conference on Electrical Engineering and Information and Communication Technology, ICEEICT 2014*, 2–6. <https://doi.org/10.1109/ICEEICT.2014.6919073>
- Krimpenis, A. A., Papapaschos, V., & Bontarenko, E. (2020). HydraX, a 3D printed robotic arm for Hybrid Manufacturing. Part I: Custom design, manufacturing and assembly. *Procedia*

- Manufacturing*, 51(2019), 103–108. <https://doi.org/10.1016/j.promfg.2020.10.016>
- KUKA. (2014). Sensitive robotics_LBR iiwa. [E-Book], 1–18.
- Liu, B., He, Y., & Kuang, Z. (2018). Design and analysis of dual-arm SCARA robot based on stereo simulation and 3D modeling. *2018 IEEE International Conference on Information and Automation, ICIA 2018, August*, 1233–1237. <https://doi.org/10.1109/ICInfA.2018.8812353>
- MATLAB. (n.d.). *MATLAB*. Retrieved October 19, 2021, from <https://www.mathworks.com/products/matlab.html>
- Mecademic. (n.d.). *No Title*. Retrieved October 21, 2021, from <https://www.mecademic.com/en/how-is-orientation-in-space-represented-with-euler-angles>
- Ngia, L. S. H., & Sjöberg, J. (2000). Efficient training of neural nets for nonlinear adaptive filtering using a recursive Levenberg-Marquardt algorithm. *IEEE Transactions on Signal Processing*, 48(7), 1915–1927. <https://doi.org/10.1109/78.847778>
- ReachRobotics. (2018). *ReachRobotics*. <https://medium.com/@ReachRobotics/the-7-robots-that-shaped-the-industry-and-the-engineers-who-created-them-4f5bd42b0681>
- RIA. (2019). Robotics Online, Robotic Industries Association. Robotics Online, Robotic Industries Association
- rigid.ink. (n.d.). *No Title*. Retrieved July 23, 2021, from <https://rigid.ink/blogs/news/175700615-petg-filament-heres-what-you-need-to-know>
- RoboCore. (2021). *No Title*. <https://www.robocore.net/tutorials/comparacao-entre-protocolos-de-comunicacao-serial>
- RoboT Unimate*. (2018). <https://robots.ieee.org/robots/unimate/>
- Rubio, J. de J. (2020). Stability Analysis of the Modified Levenberg – Marquardt Algorithm. *Ieee Transactions on Neural Networks and Learning Systems*, 32(8), 1–15.
- Sam, R., Arrifin, K., & Buniyamin, N. (2012). Simulation of pick and place robotics system using solidworks softmotion. *Proceedings of the 2012 International Conference on System Engineering and Technology, ICSET 2012*, 0–5. <https://doi.org/10.1109/ICSEngT.2012.6339325>
- Sophie Hand. (2020). *A brief history of collaborative robots*. <https://www.mhlnews.com/technology-automation/article/21124077/a-brief-history-of-collaborative-robots>

- Spatial Transformation Matrices. (n.d.). *Spatial Transformation Matrices*. Retrieved October 19, 2021, from <https://www.brainvoyager.com/bv/doc/UsersGuide/CoordsAndTransforms/SpatialTransformationMatrices.html>
- Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2006). Robot modeling and control. *IEEE Control Systems*, 26(6), 113–115. <https://doi.org/10.1109/MCS.2006.252815>
- Stansbury, J. W., & Idacavage, M. J. (2016). 3D printing with polymers: Challenges among expanding options and opportunities. *Dental Materials*, 32(1), 54–64. <https://doi.org/10.1016/j.dental.2015.09.018>
- Stilli, A., Grattarola, L., Feldmann, H., Wurdemann, H. A., & Althoefer, K. (2017). Variable Stiffness Link (VSL): Toward inherently safe robotic manipulators. *Proceedings - IEEE International Conference on Robotics and Automation*, 4971–4976. <https://doi.org/10.1109/ICRA.2017.7989578>
- Sundaram, D., Sarode, A., & George, K. (2019). Vision-Based Trainable Robotic Arm for Individuals with Motor Disability. *2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2019*, 0312–0315. <https://doi.org/10.1109/UEMCON47517.2019.8993075>
- Thingiverse. (n.d.). *WE-R2.4 Six-Axis Robot Arm by LoboCNC*. Retrieved November 23, 2020, from <https://www.thingiverse.com/thing:3327968>
- Universal Robots. (n.d.). *UR5 collaborative robot arm*. Retrieved November 23, 2020, from <https://www.universal-robots.com/products/ur5-robot/>
- Yang, H., Yan, Y., Su, S., Dong, Z., & Ul Hassan, S. H. (2019). LWH-Arm: A Prototype of 8-DoF Lightweight Humanoid Robot Arm. *Proceedings of 2019 3rd IEEE International Conference on Robotics and Automation Sciences, ICRAS 2019*, 6–10. <https://doi.org/10.1109/ICRAS.2019.8809073>
- Zacharaki, A., Kostavelis, I., Gasteratos, A., & Dokas, I. (2020). Safety bounds in human robot interaction: A survey. *Safety Science*, 127(February), 104667. <https://doi.org/10.1016/j.ssci.2020.104667>

Manual do utilizador

Manual de utilizador

Este manual guia serve para ajudar qualquer utilizador a manusear o braço robótico e também navegar e explorar os menus que existe no painel de interface do utilizador GUIDE.

Índice

Manual de utilizador	1
Descrição dos componentes em cada menu.....	3
Menu Conectar.....	3
Menu Alarme.....	5
Menu Mover	6
Menu Controlo	7
Menu Programa	10
Instrução e execução	11
Estabelecer a ligação entre PC – Matlab e o robô	11
Movimentar o braço robótico em manual	14
Efetuar a calibração	16
Gravar pontos.....	18
Alterar pontos	23
Apagar instrução (ponto)	25
Executar instrução em manual	26
Executar instrução em automático	28
Execução em modo teste	30
Gestão de alarme	31
Conclusão geral.....	33

Descrição dos componentes em cada menu

Menu Conectar

No menu “conectar” as letras representadas na figura abaixo são designadas por:



A – Botão “Refresh” permite ao programa procurar as portas COM’s conectada disponíveis.

B – Seletor ao qual permite selecionar a porta COM disponível.

C – Seletor para selecionar a taxa de transmissão (predefinido 500000 bps)

D – Botão para estabelecer a comunicação. Quando o nome for “Connect” permite estabelecer a comunicação, “Disconnect” permite fechar a comunicação.

E – Sinalizador de comunicação (verde – comunicação estabelecida; vermelho – comunicação não estabelecida).

F – Sinalizador de alarme. Verde não existe alarme, vermelho existe alarme.

bloqueando o movimento e gera as seguintes mensagens:

G – Seletor para diferente modo de funcionamento. Modo “Manu” (manual) permite mover o braço robótico de acordo com o menu “Mover”, modo “Auto” (automático) permite que o robô se

movimenta automaticamente seguindo uma sequência de pontos gravado no menu “Programa” e por fim o modo “Teste” (teste) permite que o robô se movimenta aleatoriamente para determinada posição. Este modo permite aquisição de alguns parâmetros para treinamento de rede neuronal artificial.

H – Trata de um botão e sinalizador em que quando está de cor vermelha todas ou algumas das drives estão desabilitadas, cor laranja drives habilitadas em modo manual e cor verde, drives habilitadas em modo automático/teste. Drive habilitada significa que o robô pode executar movimento e que não existe alarme exceto em manual com apenas alarme pedido de calibração.

I – O botão “Start” permite habilitar drives quando não existe alarme ou quando existe alarme de pedido de calibração e em modo manual.

J – O botão “Stop” permite desabilitar as drives dos motores, interrompendo o movimento.

K – O botão “Rearme” permite rearmar os alarmes e guardar os mesmos no histórico de alarme. Após o rearme caso o defeito não for rearmado o(s) mesmo(s) será sinalizado(s) novamente.

L – O editor de número permite definir o peso da carga útil aplicado no efector.

M – Permite escrever um comando ou selecionar a partir do botão “P” e enviar ao ESP32 principal.

N - Contém os parâmetros (número de iteração, população, erro e tempo) sendo alterável para algoritmo PSO a ser aplicado na cinemática inversa.

O – Permite visualizar, enviar e receber dados parâmetros entre a interface do utilizador e ESP32 principal.

P – Contém os valores de comprimentos de cada elo.

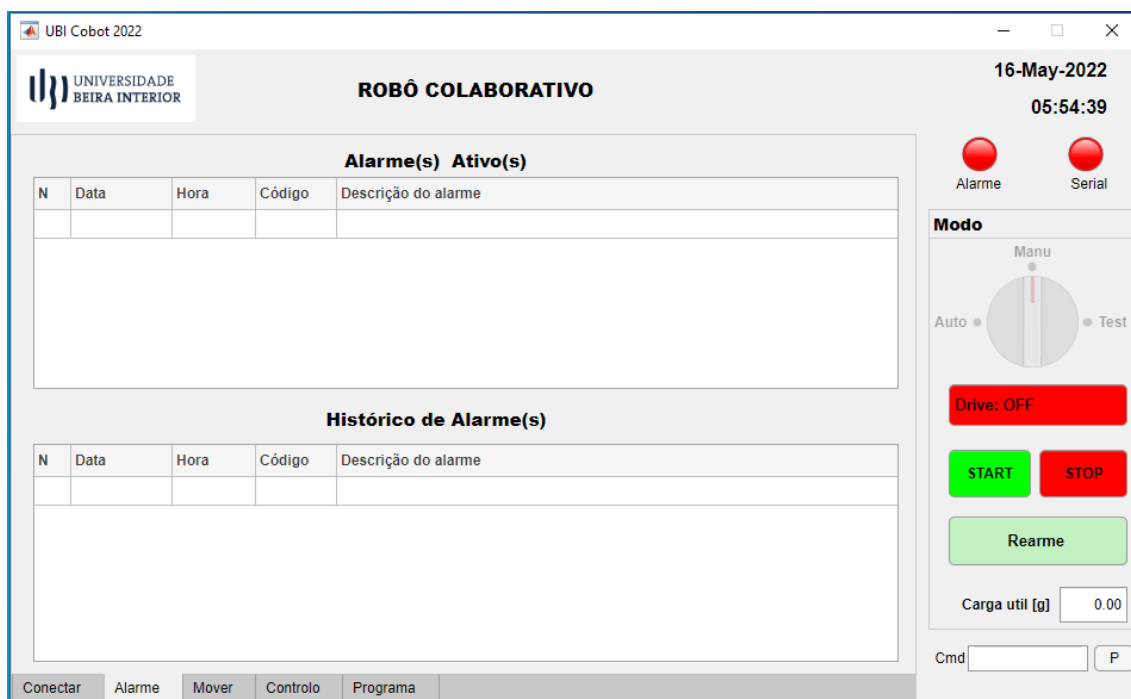
Q – Contém os valores do limite de ângulo para cada juntas.

R – Contém o valor da altura mínima de segurança da base do robô até a ponta de cada elo e também a distância mínima de segurança entre centro da base do robô até a ponta de cada elo. Sempre que a posição de cada elo não satisfizer esses parâmetros, o robô entra em segurança, gerando alerta.

S – Símbolo da UBI e com a funcionalidade de um botão que quando for pressionado, força a paragem do programa fechando a interface do utilizador.

Menu Alarme

O menu alarme apresentado na figura abaixo, permite sinalizar o(s) alarme(s) ativo(s) após o seu surgimento e também o historial de alarmes.



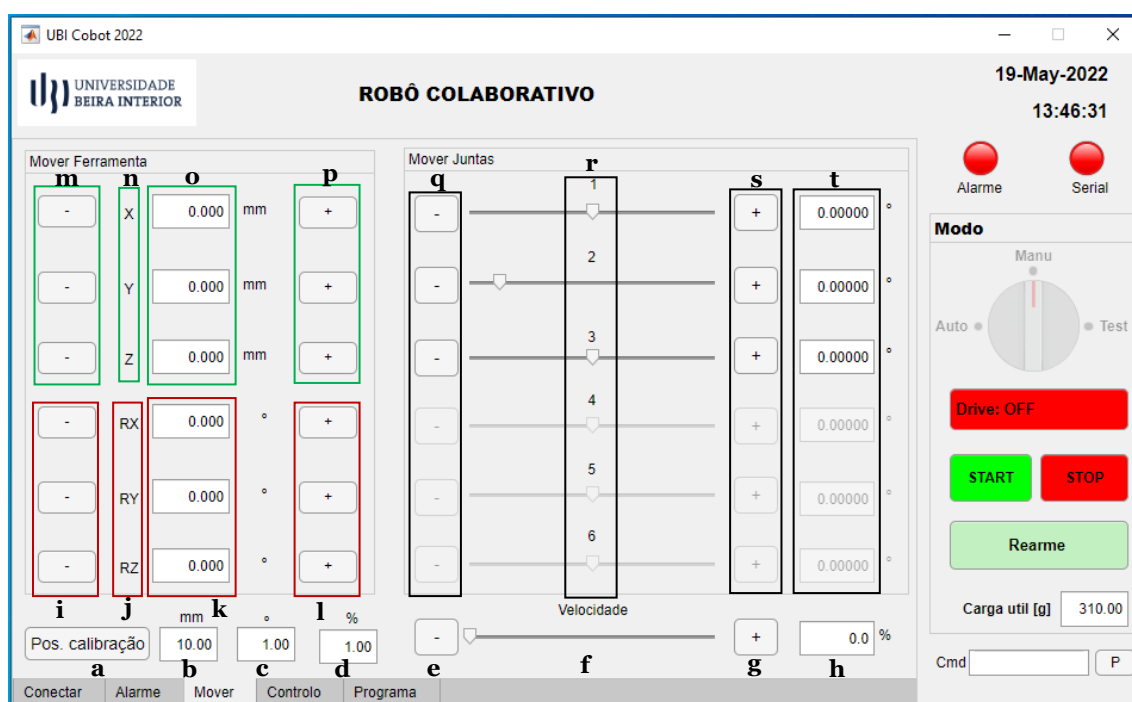
Os alarmes são sinalizados numericamente pela ordem de acontecimento, com a descrição, código, data e hora do acontecimento. Na tabela de histórico, permite visualizar o historial dos alarmes apresentando o número do alarme, código, data e hora. Quando existe alarme(s), após clicar no botão rearme, o(s) alarme(s) ativo(s) serão salvo no historio, atualiza o ficheiro de texto do histórico (historico_cobot.txt) existente no arquivo do programa e limpa a tabela de alarme(s) ativo(s).

Todos os alarmes são codificados com um número. Quando o ESP32 principal deteta ou receber do ESP32 de cada eixo o(s) código(s) do(s) alarme(s), os códigos dos alarmes são enviados para a interface do utilizador (PC - MATLAB) para ser sinalizado. Os alarmes são sinalizados apenas uma vez até os mesmos serem rearmados.

Os códigos e as descrições dos alarmes estão armazenados num ficheiro de texto (alarmes_cobot.txt) nos arquivos do programa de interface, sendo que os mesmos podem ser editados fora do programa MATLAB. O ficheiro alarmes_cobot.txt é acedido apenas uma vez apos a inicialização do programa de interface e o ficheiro historico_cobot.txt é acedido no inicio e sempre que for feito o rearme e existir alarme(s) ativo(s). Caso esses ficheiros não existam, ocorrerá um erro não visível na abertura dos mesmos o que levará à sinalização dos alarmes ativos ou historial a sem descrição dos mesmos.

Menu Mover

O menu alarme representado na figura abaixo, permite manusear o braço robótico em manual, alterar a velocidade de execução, alterar os valores a incrementar/decrementar e também permite mover o mesmo para a posição de calibração.



Os botões e os campos editáveis estão identificados por letras em que:

- a – Botão “Pos. calibração” para mover o braço robótico para posição de calibração.
- b – Editor de número que contém o valor a ser incrementado/decrementado na posição do efector.
- c - Editor de número que contém o valor a ser incrementado/decrementado na orientação do efector.
- d - Editor de número que contém o valor a ser incrementado/decrementado na percentagem da velocidade de execução.
- e – Botão que permite incrementar o valor contido em “d” na percentagem de velocidade de execução.
- f – Botão deslizante (slider) que permite varia diretamente a percentagem da velocidade de execução de 0 a 100.

g – Botão que permite incrementar o valor contido em “d” na percentagem de velocidade de execução.

h – Editor de número que permite definir diretamente a percentagem da velocidade de execução.

Nota 1: Quando a percentagem da velocidade de execução muda, “f” e “h” mudará automaticamente.

i – Permite decrementar o valor contido no “c” a orientação da manipulado consoante o eixo “j”.

j – Eixo (x, y e z) de orientação do manipulador.

k – Editor de número que permite mudar diretamente a orientação da ferramenta mediante o eixo “j”.

l – Permite incrementar o valor contido no “c” a orientação da manipulado consoante o eixo “j”.

m – Permite decrementar o valor contido no “b” a posição do efector consoante o eixo “n”.

n – Eixos (x, y e z) de posição do efector.

o – Editor de número que permite mudar diretamente a posição da ferramenta mediante o eixo “n”.

p – Permite incrementar o valor contido no “b” a posição do efector consoante o eixo “n”.

q – Botões que permite decrementar o valor contido no “c” ao angulo para cada eixo (1, 2, ... 6).

r – Botões deslizantes (slider) que permite mudar a angulo de cada eixo pretendido automaticamente.

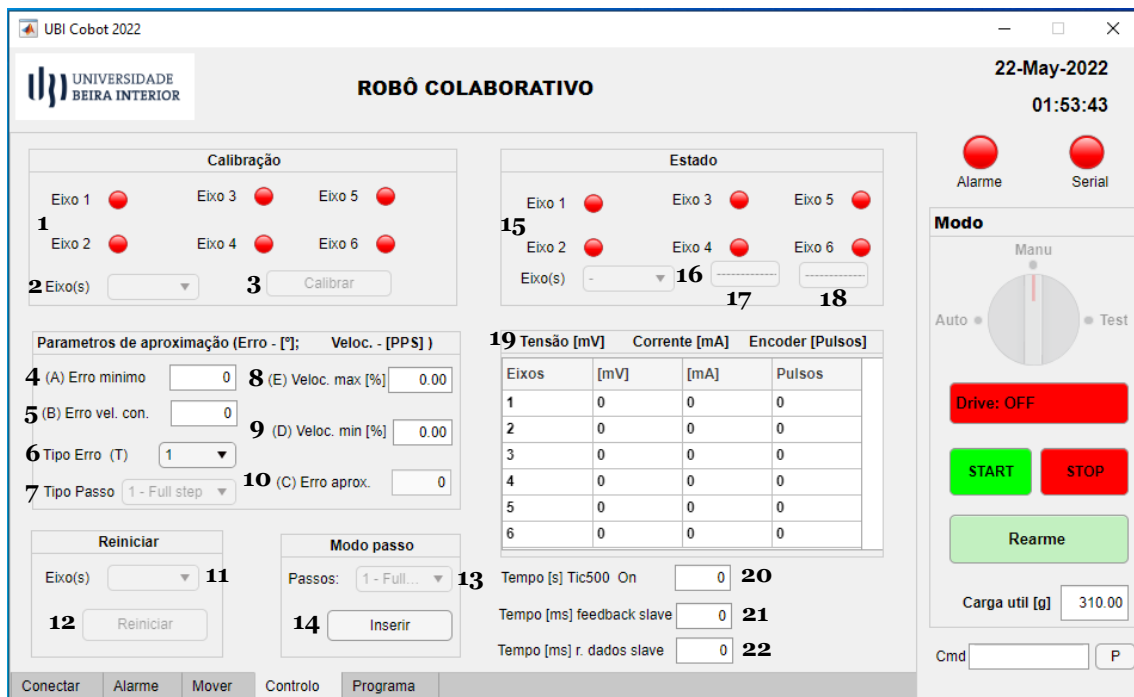
s – Botões que permite incrementar o valor contido no “c” ao angulo para cada eixo (1, 2, ... 6).

t – Editores de número que permite mudar diretamente o angulo pretendido para cada junta.

Nota 2: Os valores de “k, o, r e t” são proporcionais e qualquer alteração, estes são atualizados automaticamente.

Menu Controlo

O menu designado por “Controlo” representado na figura abaixo, permite visualizar, sinalizar, editar e seleccionar parâmetros.



O grupo designado por **calibração** contém:

- 1 – Sinalizador de estado de calibração para cada eixo. Este sinalizador pode ser de cor vermelho (eixo não calibrado) e verde (eixo calibrado).
- 2 – Caixa de lista que permite seleccionar o(s) eixo(s) (individual ou todos) para calibrar.
- 3 – Botão para enviar pedido de calibração. Este botão será desbloqueado automaticamente após a seleção de eixo(s) (2).

O grupo designado por **parâmetros de aproximação** contém parâmetros quando o braço robótico estiver a aproximar-se ao ângulo pretendido durante o movimento. Assim sendo:

- 4 – Parâmetro editável para erro mínimo pretendido.
- 5 – Parâmetro editável, erro mínimo para velocidade mínima constante.
- 6 – Permite seleccionar tipo de erro pretendido sendo erro simples ou quadrado.
- 7 – Permite seleccionar tipo passo aplicar ao motor.
- 8 – Parâmetros editável com a percentagem da velocidade máxima de aproximação.
- 9 – Parâmetros editável com a percentagem da velocidade mínima de aproximação.

10 – Parâmetros não editável, contem o valor de erro quando o manipulador entra em aproximação. Este é calculado automaticamente quando se altera a percentagem da velocidade de execução.

O grupo designado **reiniciar** contém:

11 – Caixa de lista que permite seleccionar o(s) eixo(s) (individual ou todos) para reiniciar.

12 – Botão para enviar pedido para reiniciar. Este botão será desbloqueado automaticamente após a seleção de eixo(s) (11).

O grupo designado **modo de passo** contém:

13 – Caixa de lista que permite seleccionar tipo de passo aplicar ao motor durante a execução fora de aproximação.

14 – Botão para enviar tipo de passo seleccionado no 13. Este botão será desbloqueado automaticamente após a seleção de passo (13).

O grupo designado por Estado contém os estados dos controladores dos motores em que:

15 – Sinalizador dos estados de controlador de motor em que, vermelho drive desabilitada, verde drive habilitada, ou seja, o eixo referente reúne condições para entrar em execução e azul drive habilitado e motor em funcionamento ou veio de bloqueado.

16 – Caixa de lista que permite seleccionar o eixo a bloquear ou desbloquear o veio do motor.

17 – Botão para bloquear o veio do motor de eixo seleccionado no 16.

18 – Botão para desbloquear o veio do motor de eixo seleccionado no 16.

19 – Tabela não editável em que contém o valor da tensão, corrente elétrica e número de pulsos do encoder de cada eixo.

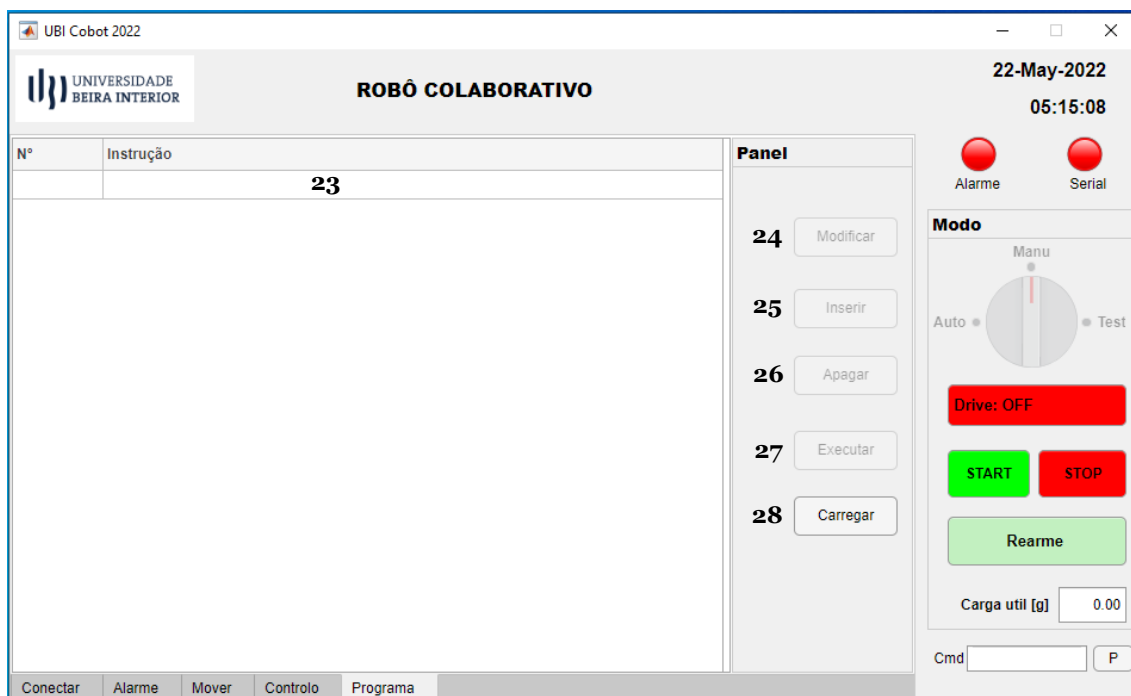
20 – Parâmetros editável em que contém o valor de tempo para o veio do motor ficar preso após o movimento.

21 – Parâmetros editável em que contém o valor de tempo máximo que o ESP32 principal espera a resposta dos ESP32 de cada eixo.

22 – Parâmetros editável em que contém o valor de tempo máximo que o ESP32 principal espera a receber dados dos ESP32 de cada eixo.

Menu Programa

O menu Programa é representado na figura abaixo.



Em que o número:

23 – Tabela que contém as instruções para cada ponto gravado.

24 – Botão que permite modificar um ponto contemplado na tabela (23).

25 – Botão que permite inserir uma instrução na linha seguinte na tabela (23).

26 – Botão que permite apagar uma instrução na tabela (23).

27 – Botão que permite executar individualmente instruções da tabela (23).

28 – Botão que permite carregar instruções (programa) para a tabela (23).

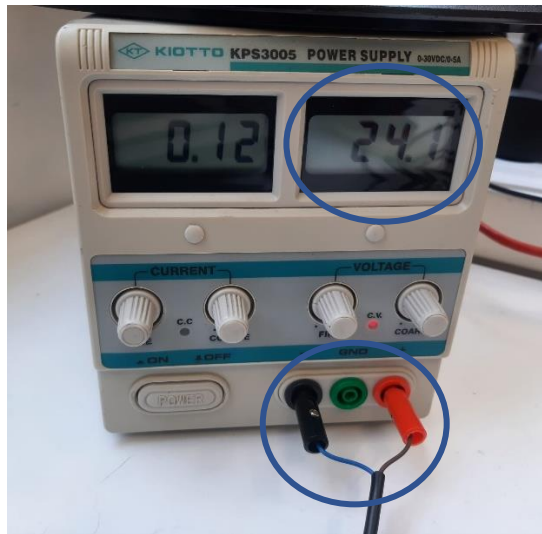
Instrução e execução

Neste capítulo apresentam-se os procedimentos passo a passo para:

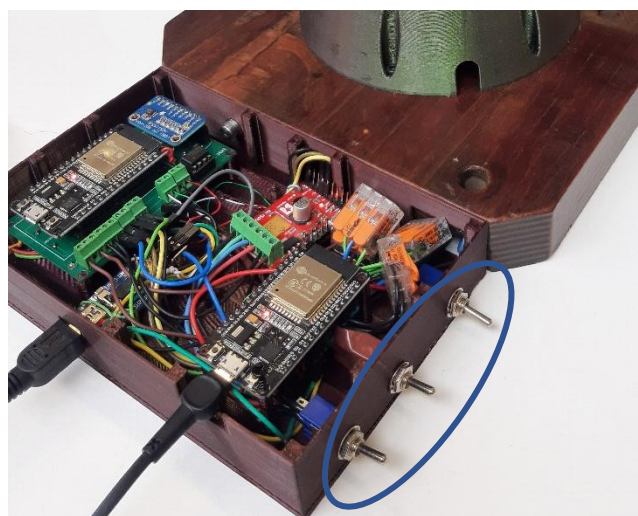
Estabelecer a ligação entre PC – Matlab e o robô

- 1) Ligar eletricamente o ESP32 principal, os ESP32 secundário de cada eixo e as drives do motor, para tal é preciso:

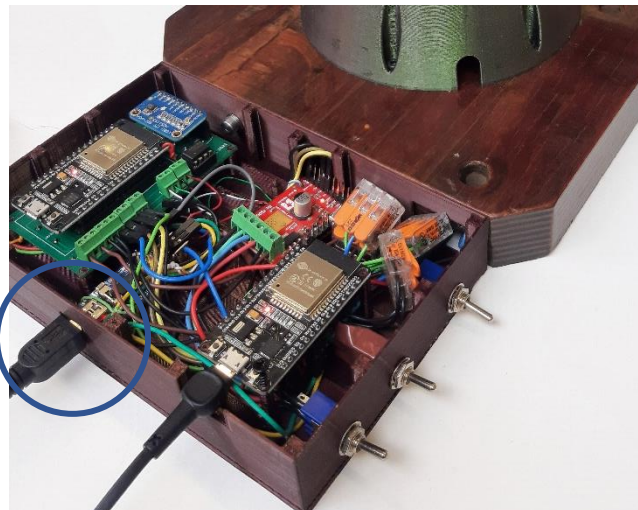
- Alimentação elétrica principal de 24 V



- Comutar os 3 switch para “ON” voltado o interruptor para cima. Os switch encontra-se na caixa na base do robô.

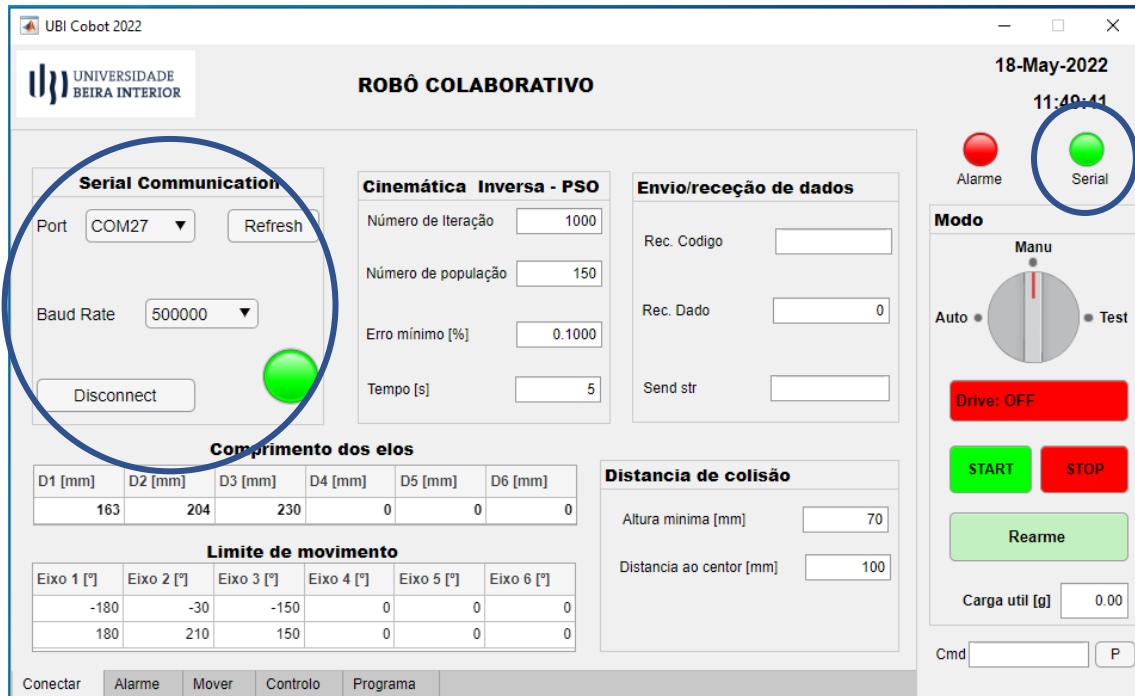


- 2) Estabelecer a ligação física através do cabo USB/USB mini entre o conversor (base do robô) e o PC.



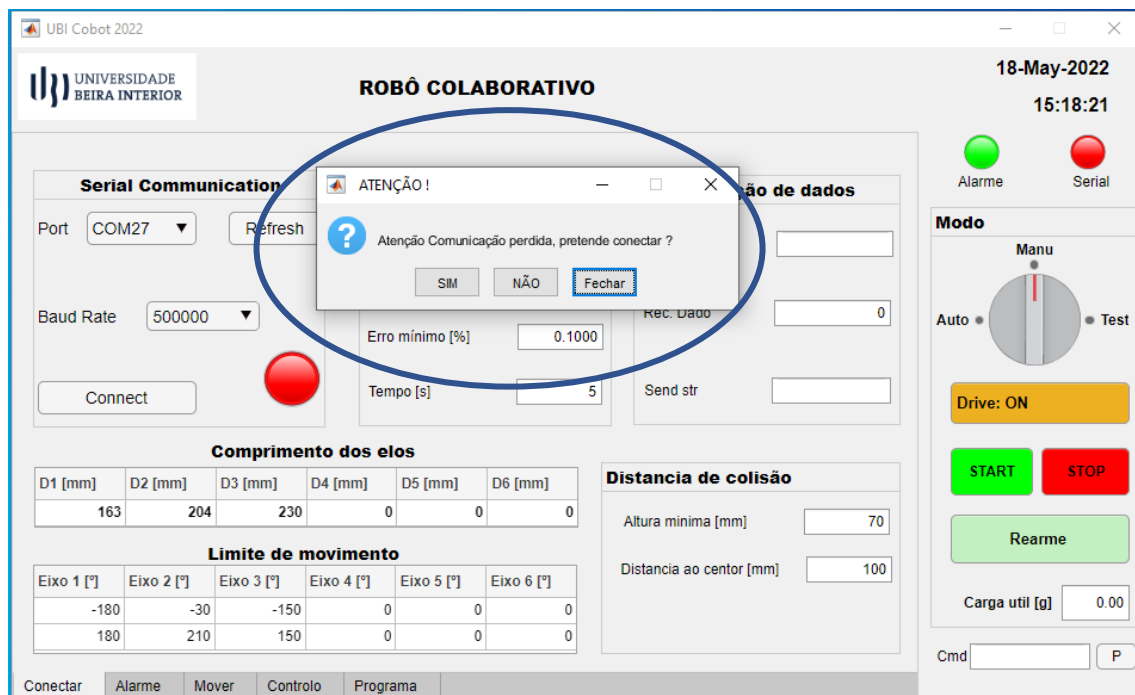
- 3) Abrir o programa de interface do utilizado, selecionar o menu “Conectar “ (figura abaixo) e estabelecer a comunicação clicando no botão “Refresh” “A”, selecionar a porta USB “B” ao qual foi ligado o cabo, selecionar “Baud rate” “C” para 50000 e por fim clicar no botão “Connect” “D”.

Quando a comunicação entre GUIDE – PC e robô for estabelecida (figura abaixo) a sinalização “E” sinalizará com a cor verde.



A comunicação é interrompida quando:

- For clicado no botão “D” “Disconnect”.
- Se o cabo USB for desconectado aparecendo um alerta conforme a figura abaixo passado 5 segundo.



A opção, “Fechar” permite fechar a mensagem e o painel de interface, “Não” permite fechar a mensagem e não estabelece a comunicação e a opção “SIM” permite restabelecer a comunicação se o cabo USB for ligado na mesma entrada USB antes de ser desconetado, caso contrário deve seleccionar opção “NÃO” e repetir o passo 3 para restabelecer a comunicação.

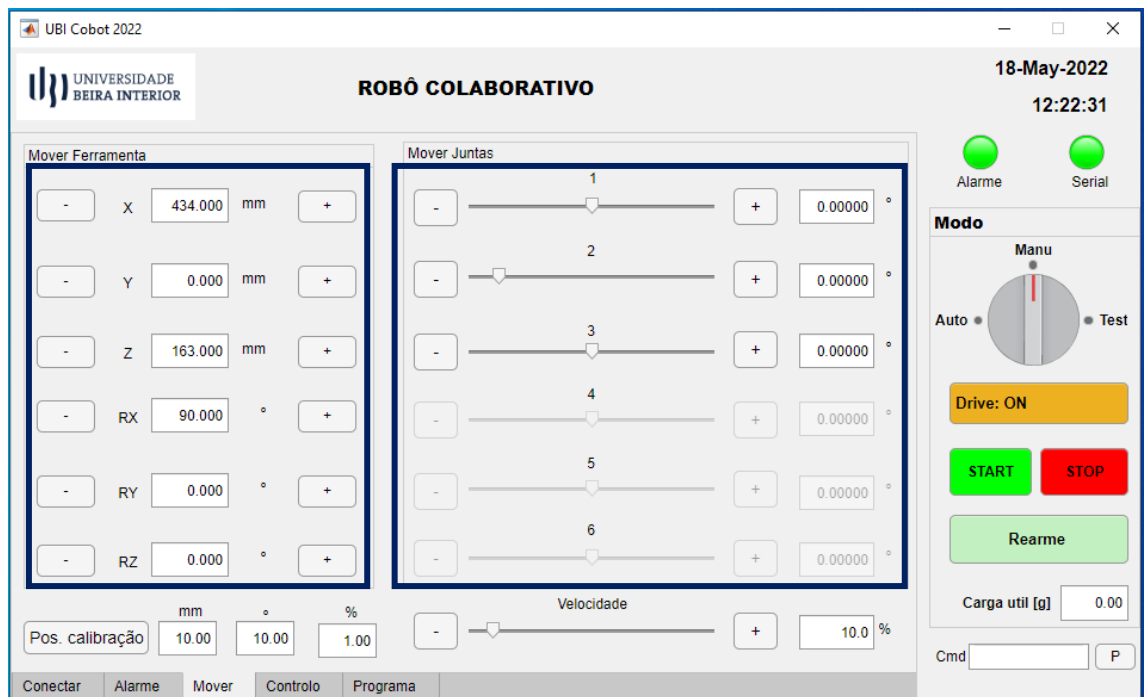
Movimentar o braço robótico em manual

Após estabelecer a comunicação entre o robô e o programa de interface, deve:

- 1) Se existe alarme (Sinalizador de alarme cor vermelho) exceto o pedido de calibração, deve rearmar o(s) alarme(s).
- 2) Rodar a seletor para modo manual e habilitar drive premindo o botão “START”.
- 3) Verificar se a(s) drives estão habilitadas estando com sinalização diferente de vermelho (menu controlo).



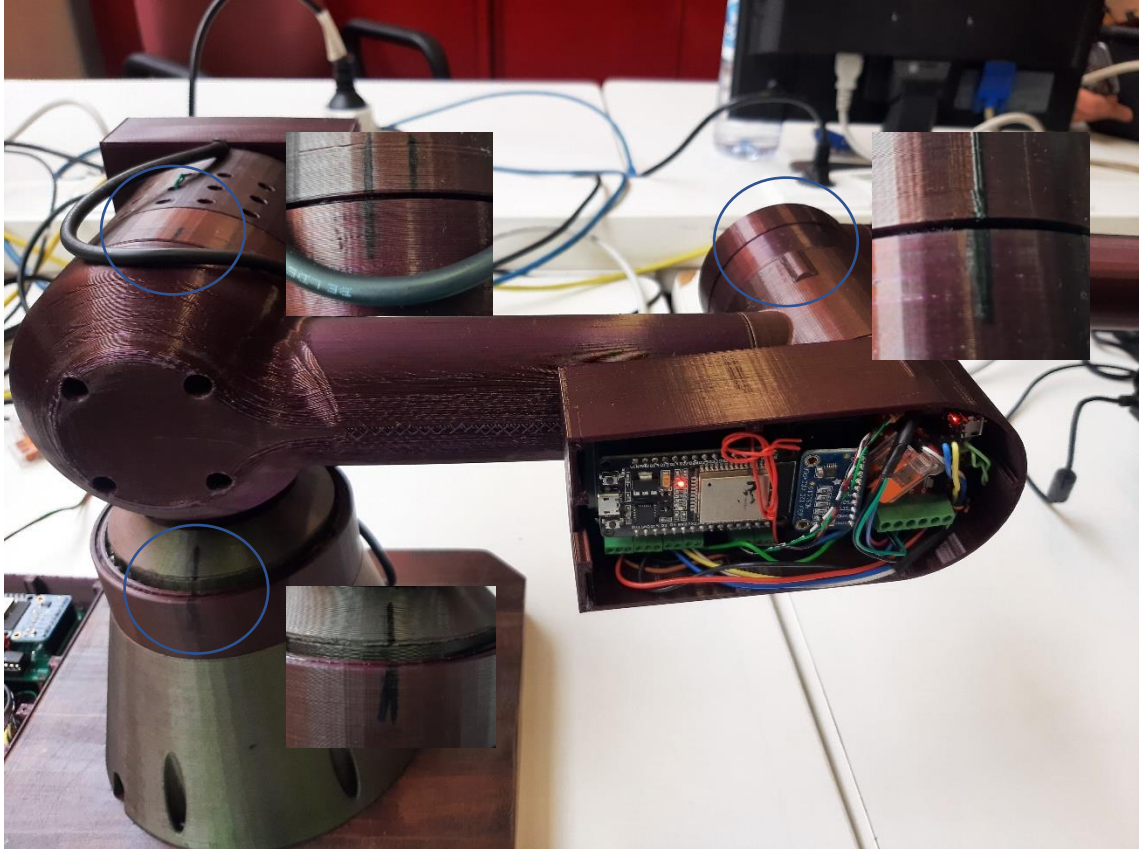
- 4) Aceder ao menu “Mover” e alterar o valor para a posição pretendida através das componentes i, k, l, m, o, p, q, r, s ou t (botão “-”, “+”, botão deslizante (slider) ou editor de posição ou ângulo).



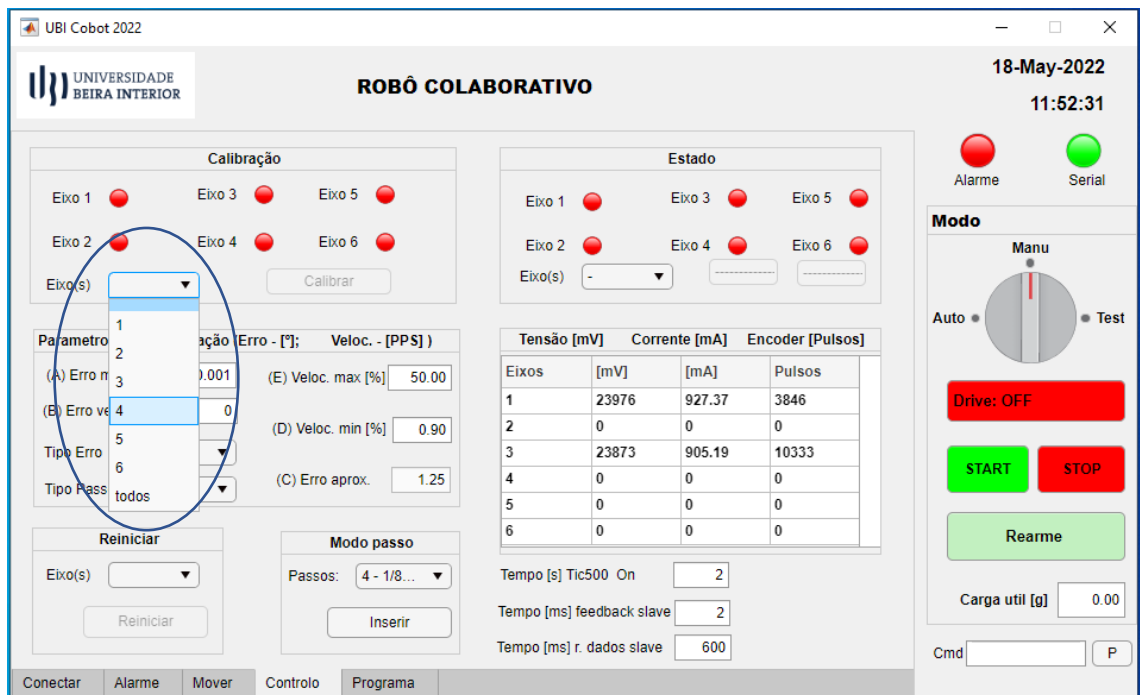
- 5) Após alterar o valor da posição pretendida, o robô irá mover automaticamente para o ângulo ou posição pretendida.

Efetuar a calibração

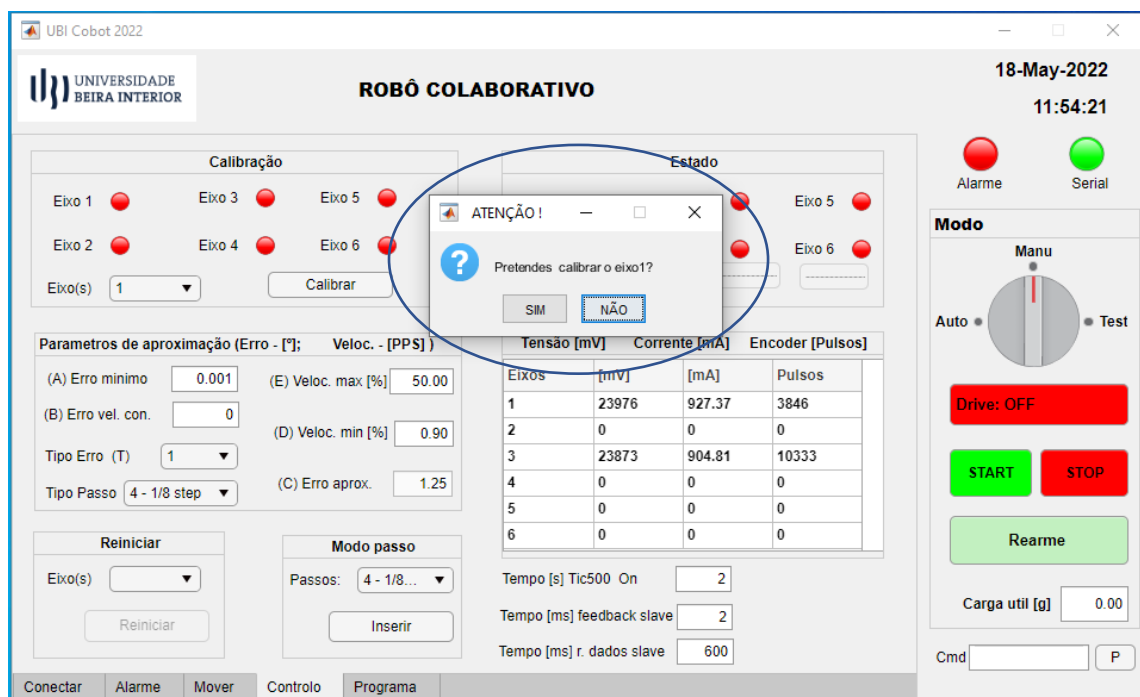
- 1) Mover o robô manualmente (Ver o procedimento movimentar o braço robótico em manual) para a posição correspondente as referências destacadas de acordo com a figura abaixo:



- 2) Aceder ao menu controlo e seleccionar o(s) eixo(s) pretendido de acordo com a figura abaixo.



- 3) Após seleccionar o eixo, o botão calibrar desbloqueará automaticamente. Deve clicar no botão calibrar e aparecerá uma mensagem para confirmar a operação conforme imagem abaixo.

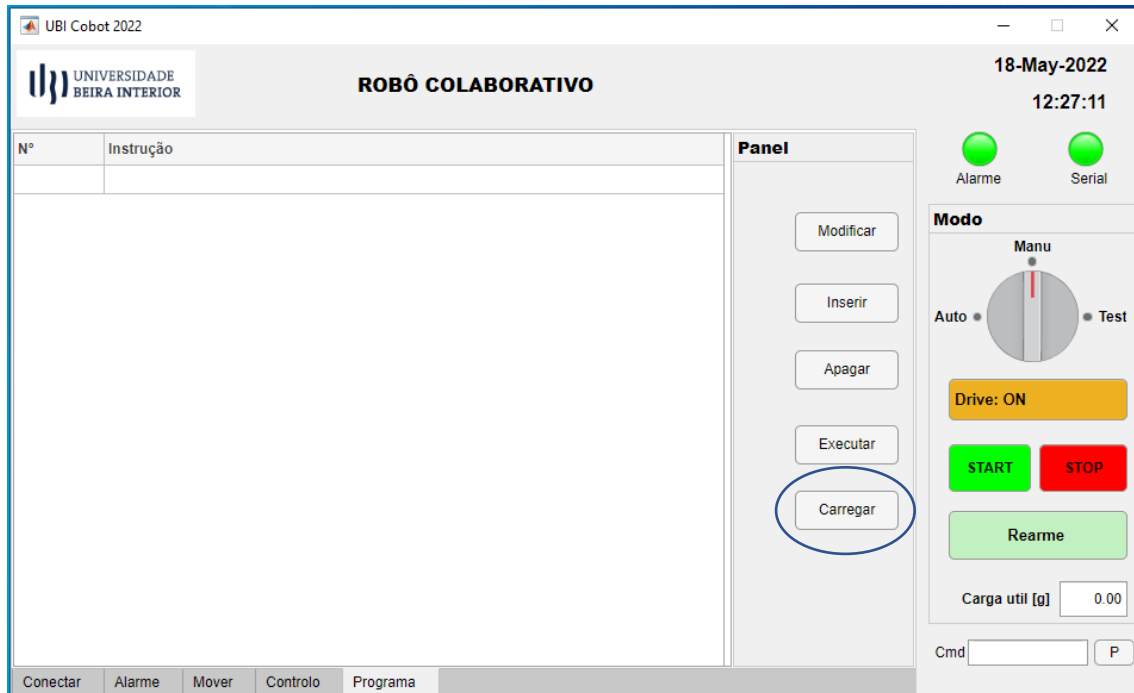


- 4) Confirmar a operação clicando no botão “SIM” ou “NÃO”. Após a confirmação de calibração, o estado do eixo calibrado passará a cor verde e se antes existia o alarme de calibração, será preciso fazer o rearme clicando na tecla “Rearme” para rearmar o defeito.

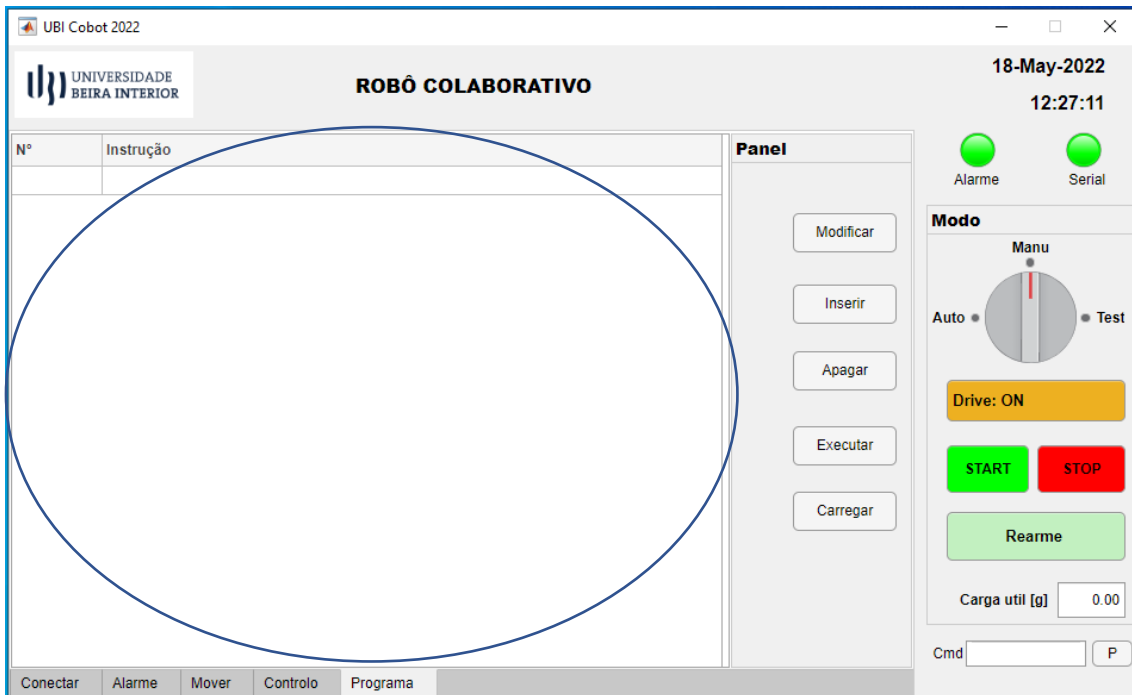
Gravar pontos

Os pontos são salvos no ficheiro de texto (prog_o.txt) no arquivo de painel de utilizador, para gravar uma posição é preciso existir a conexão entre o robô e o PC, o seletor em modo manual e também os eixos estarem calibrados. Os procedimentos para gravar um ponto são:

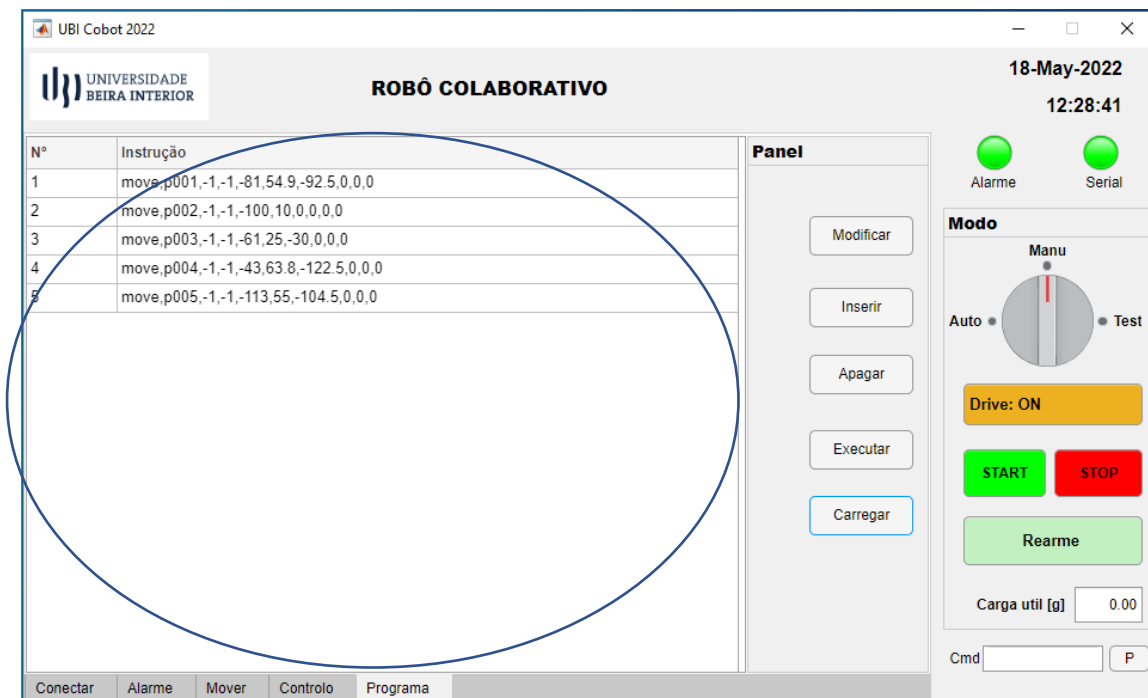
- 1) Aceder ao menu “Programa” e clicar na tecla “Carregar” para carregar o ficheiro.



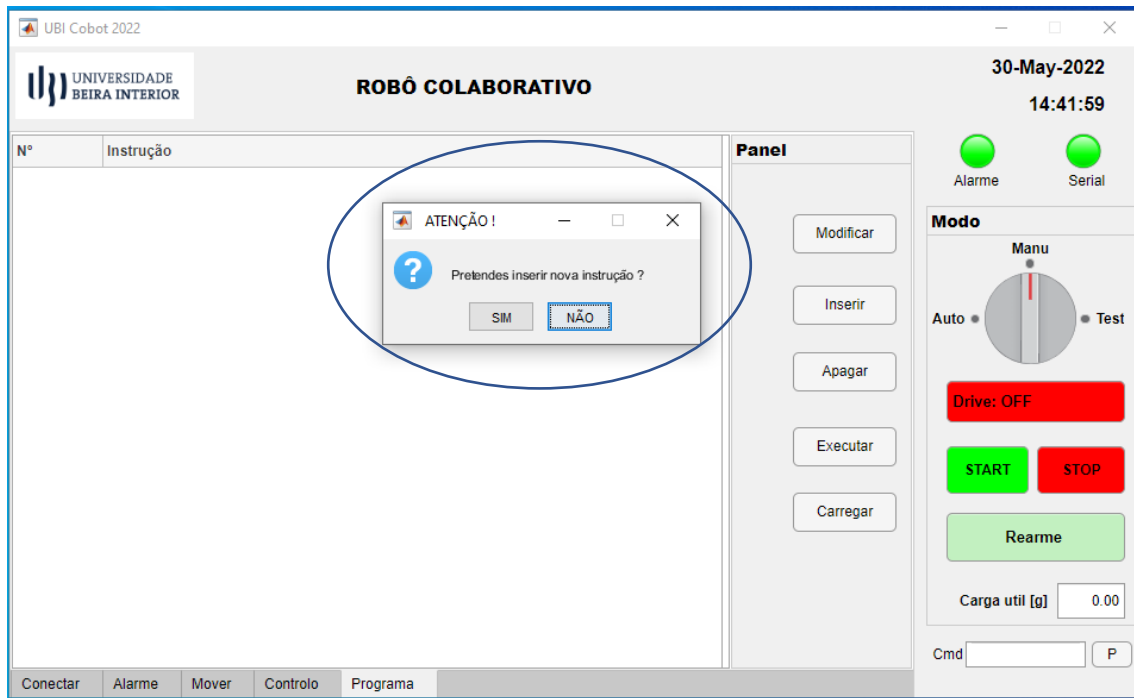
- 2) Após carregar o ficheiro se ainda não foi gravado nenhum ponto, a tabela aparecerá vazia conforme a figura abaixo.



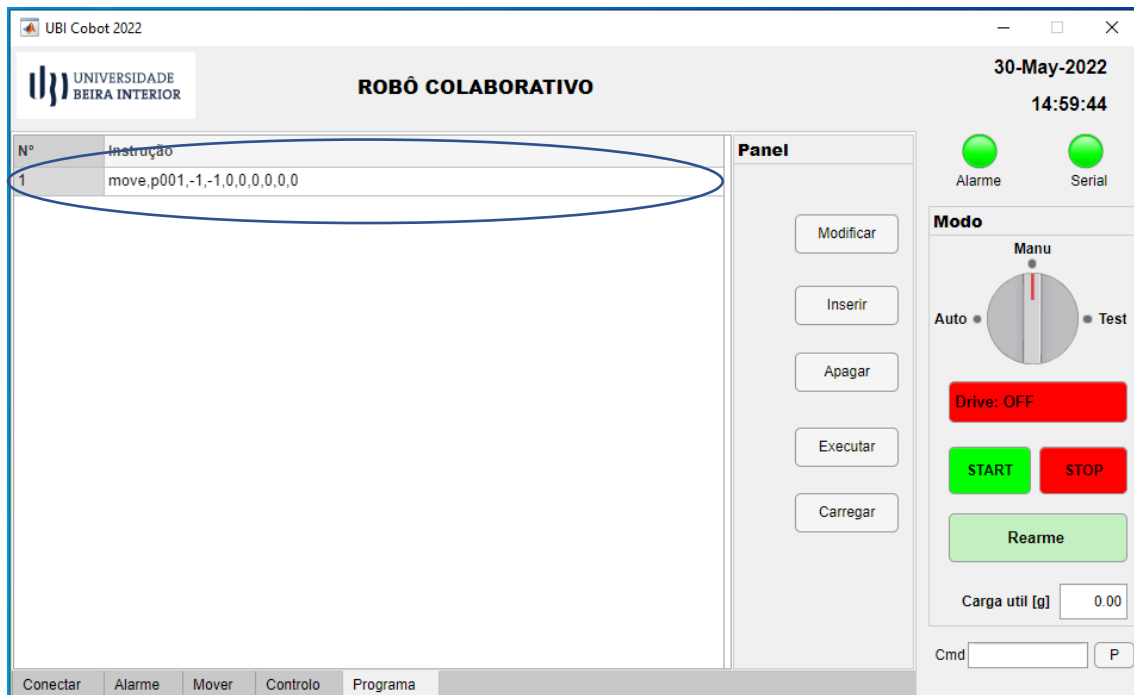
Caso já foi gravado algum ponto, a tabela estará preenchida conforme a figura abaixo.



3.1) Para gravar um ponto sem antes existir pontos gravado, deve clicar no botão “Inserir” e aparecerá uma mensagem automaticamente conforme a figura abaixo de confirmação.



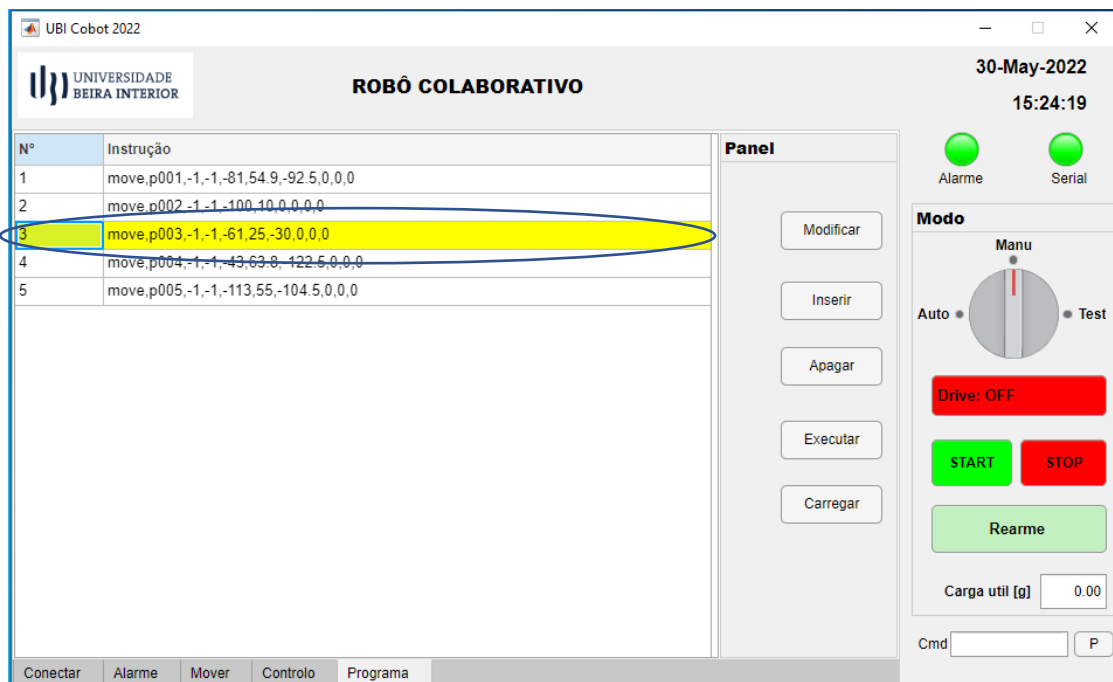
3.2) Após a confirmação “SIM”, uma instrução é inserida na tabela automaticamente conforme a figura abaixo.



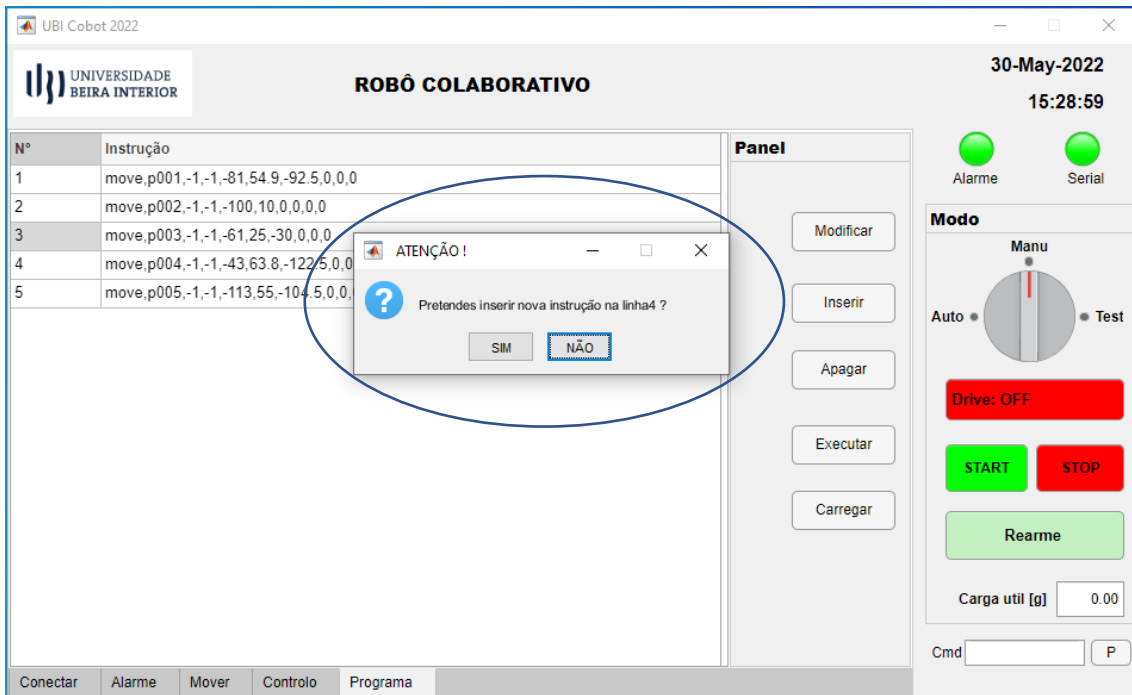
As instruções são numeradas automaticamente, começando com “move”, de seguida o número de ponto “p001” automaticamente, de seguida o 1º “-1” quer dizer que o robô se movimentará com a

percentagem de velocidade atual (parâmetro h no menu Mover), de seguida o 2º “-1” quer dizer que o robô se movimentará com o passo atual do motor atual selecionado (parâmetro 13 no menu Controlo) por os últimos 6 números corresponde o ângulo de cada junta. A posição gravada será os valores atual de cada juntas.

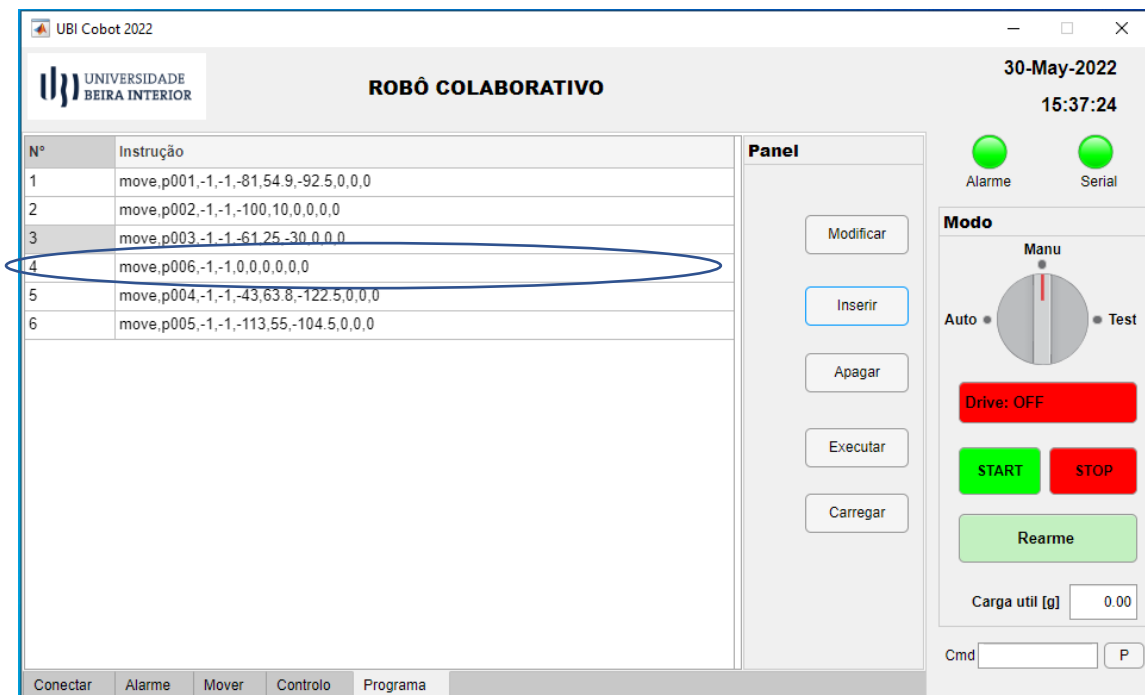
4.1) Para gravar um ponto caso já existe outros pontos gravado, deve seleccionar a linha da tabela clicando com o rato sobre a linha pretendida (a linha seleccionada ficará em amarelo) conforme a figura abaixo.



4.2) Clicar no botão “Inserir” e de seguida aparecerá uma mensagem de confirmação conforme a figura abaixo.



4.3) Após a confirmação “SIM” uma nova instrução será inserida. A nova instrução será inserida na linha a seguir a selecionada deslocando as posteriores instruções para baixo caso existe e o número da posição será numerado consoante a que já existe. Podemos verificar as mesmas na figura abaixo.

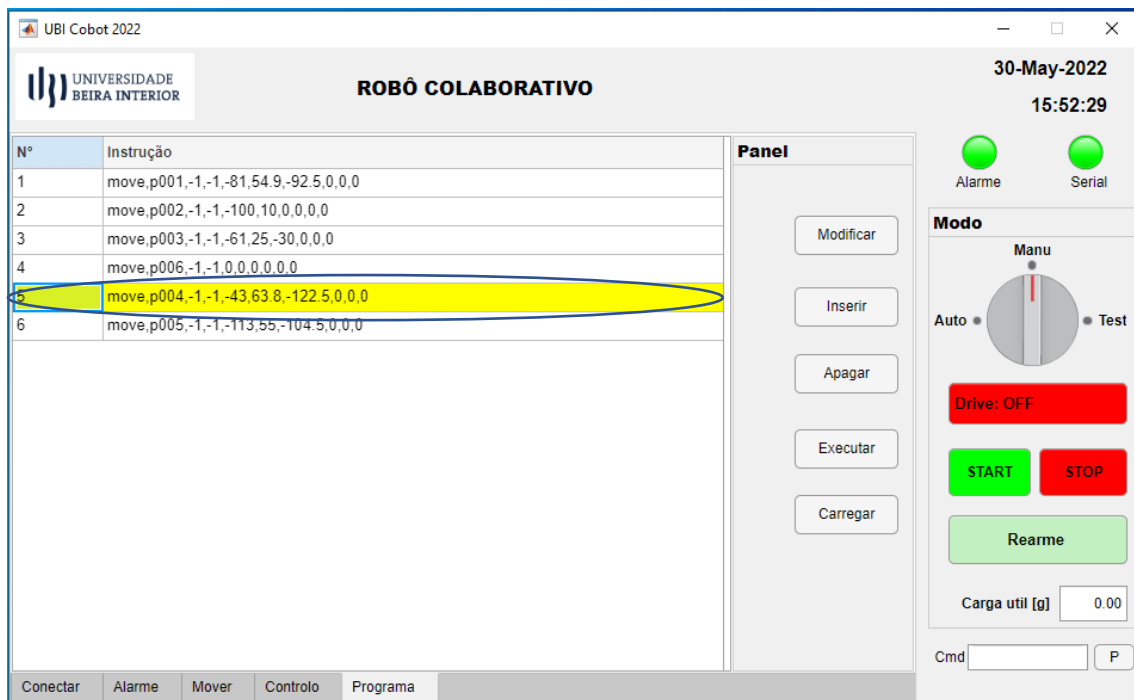


Alterar pontos

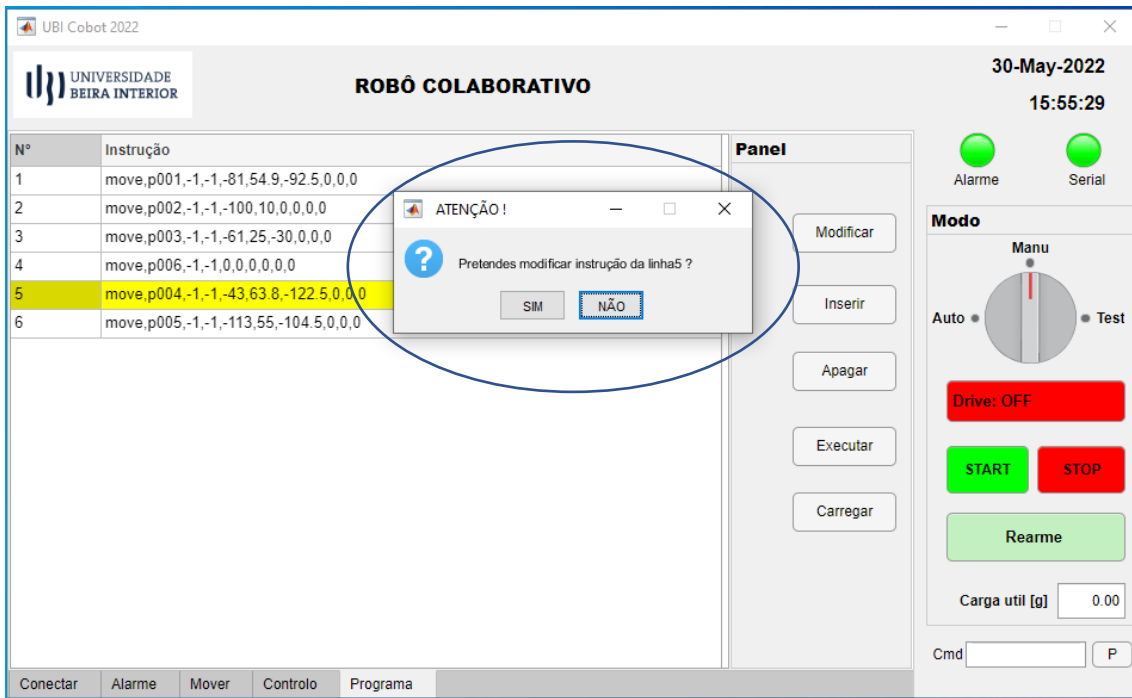
Existe duas maneiras para alterar ponto ou parâmetro de execução. As opções são:

- A primeira opção é:

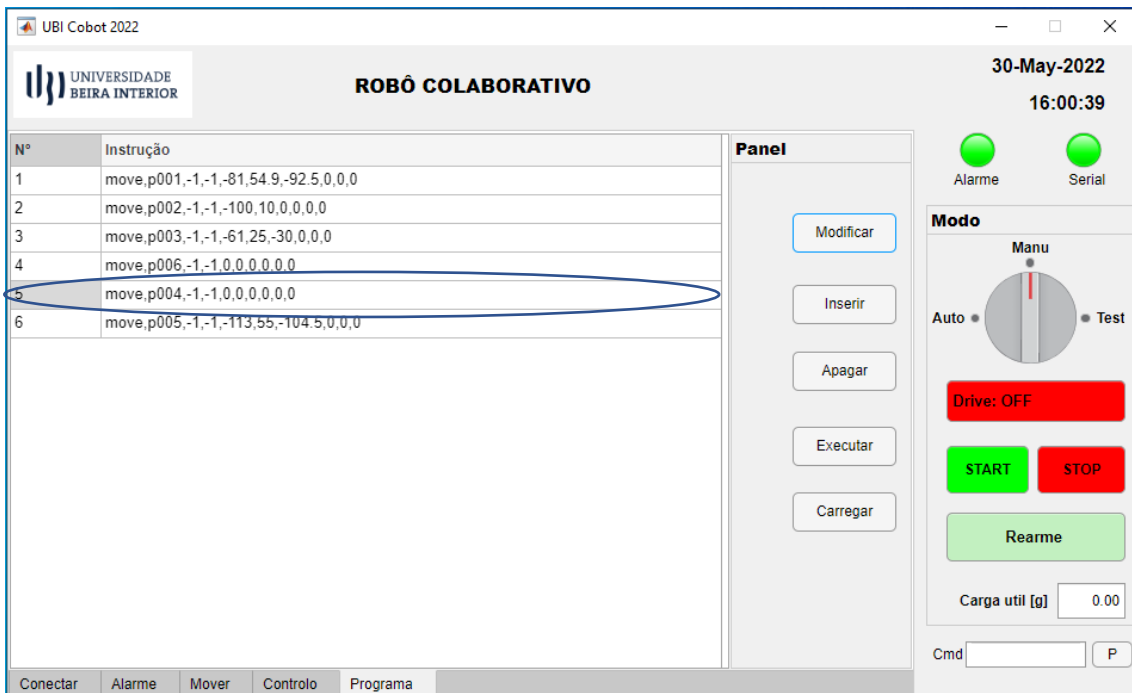
1.1) Mover o robô para a posição pretendida de seguida seleccionar a linha com a posição a alterar (linha seleccionada destacado em amarelo) conforme a figura abaixo.



1.2) Clicar no modificar e de seguida aparecerá uma mensagem para confirmação da alteração conforme a figura abaixo.



1.3) Após a confirmação “SIM” a linha selecionada (5) será alterada apenas os valores dos ângulos de cada juntas para os valores atuais conforme a figura abaixo.



- A segunda opção é:

2.1) Clicar duas vezes na instrução pretendida e editar manualmente. É possível alterar o número de posição com 3 dígitos (pxxx), a percentagem da velocidade de 0 a 100 e para manter colocar -1, o passo do motor de 1 a 4 (1 - completo, 2 - 1/2, 3 - 1/4 e 4 - 1/8) e para manter colocar -1, de seguida alterar os 6 valores de ângulos de cada juntas dentro do limite de parâmetro O no menu Conectar.

2.2) Após alteração manual, deve clicar na tecla ENTER do PC. Será feito um controlo automaticamente se a linha alterada encontra dentro dos conforme caso contrário a linha retomará a instrução inicial com os parâmetros antes de ser alterado.

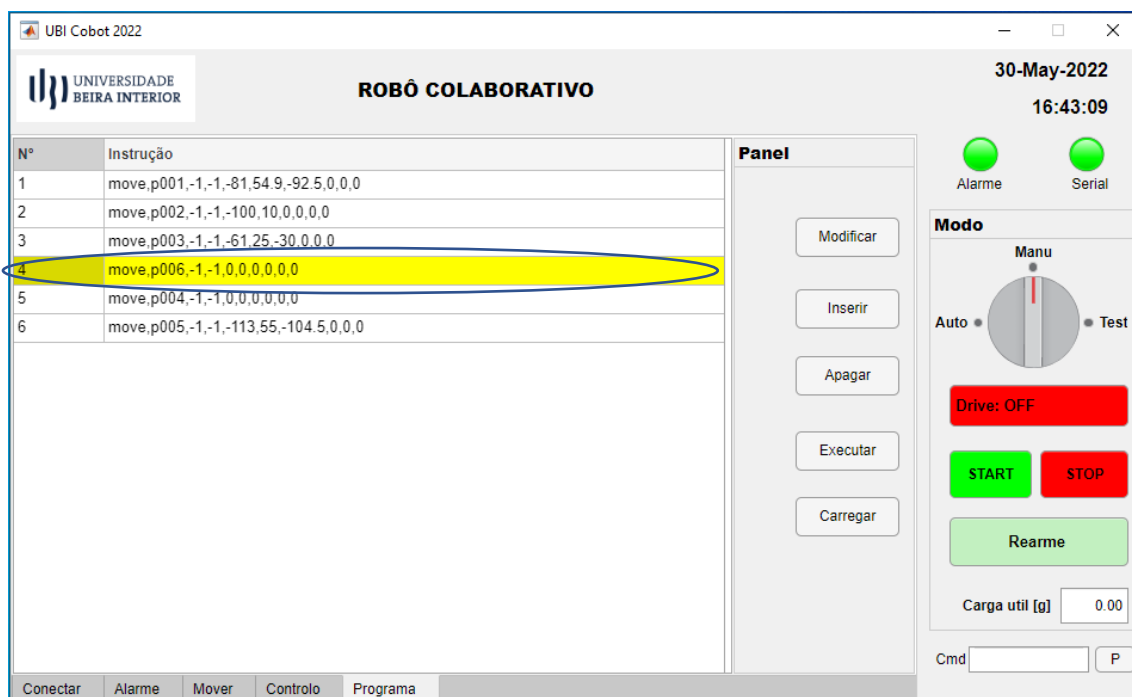
Por exemplo, dada a instrução

move,p004,-1,-1,0,0,0,0,0,0

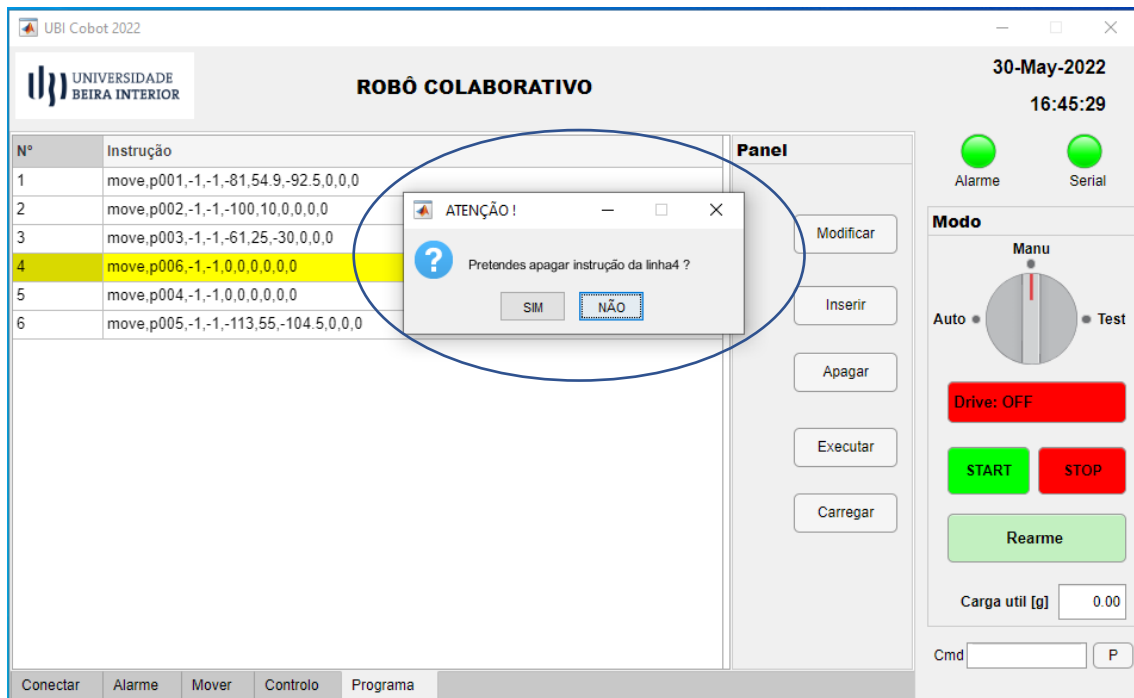
Nunca alterar o nome “move”, deve sempre conter virgulas para separar os parâmetros, o parâmetro número da posição deve sempre conter a letra p de seguida os 3 dígitos, o 1º -1 só deve conter o numero -1 ou de 0 a 100 (velocidade), no 2º -1 só deve conter o numero -1 ou de 1 a 4 (passo do motor) inteiro de seguida os valores dos 6 ângulos das juntas sempre separados de virgula. Caso contrário a instrução será considerada invalidade. Nunca usar este procedimento para apagar a instrução.

Apagar instrução (ponto)

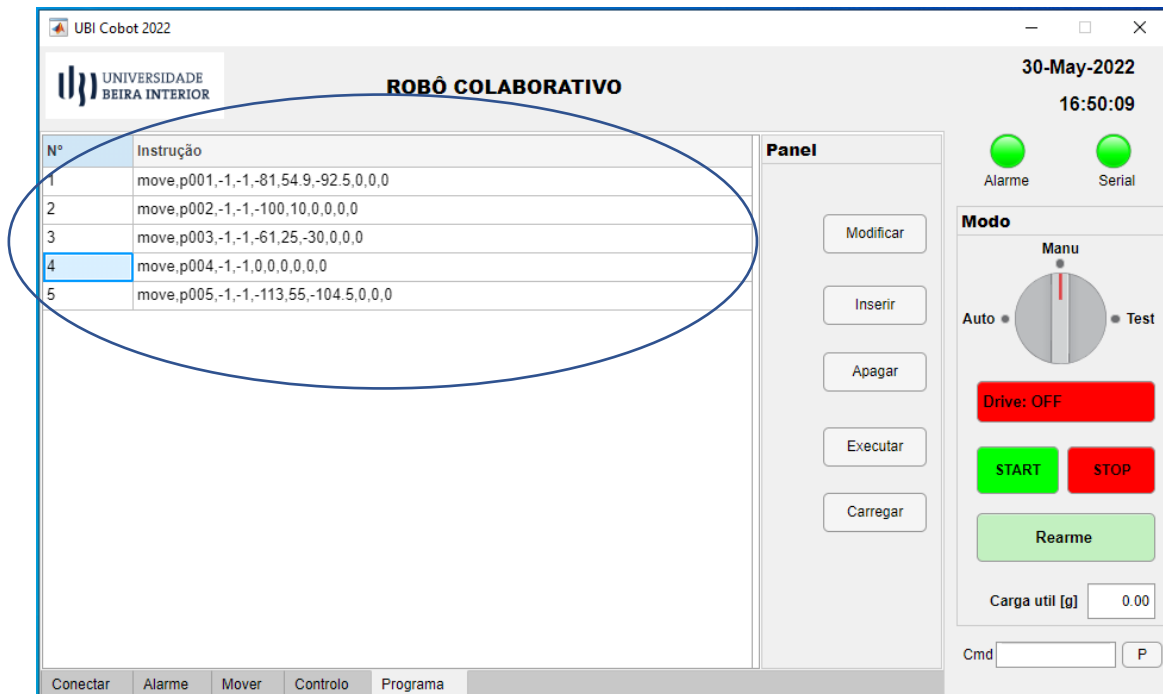
1.1) Selecionar a linha da instrução a apagar conforme a figura abaixo.



1.2) Clicar no botão “Apagar” e de seguida confirmar alteração conforme a figura abaixo.



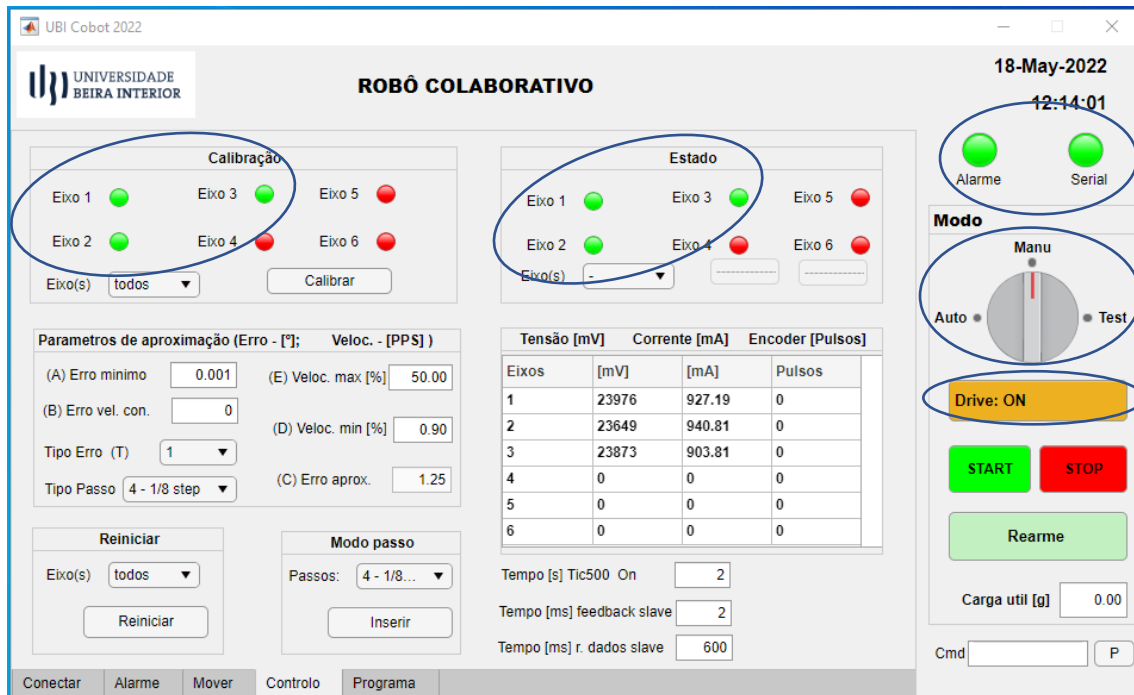
1.3) Após a confirmação, a linha selecionada será excluída automaticamente e as restantes instruções será inumerada automaticamente. Podemos averiguar esta alteração na figura abaixo.



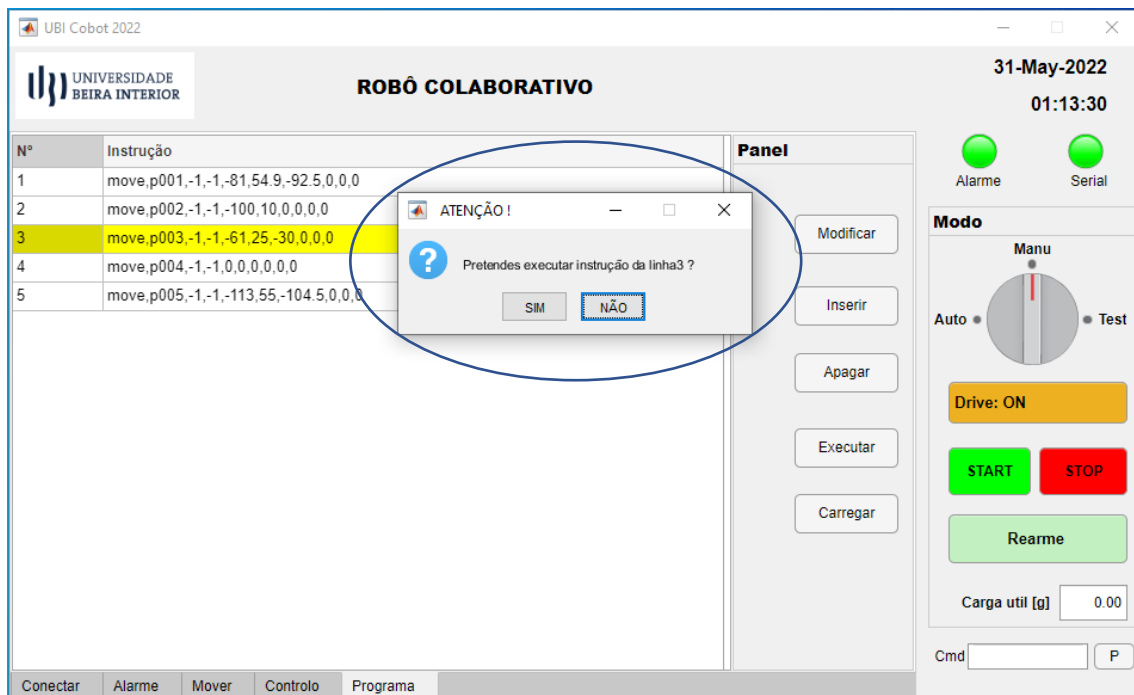
Executar instrução em manual

Este procedimento permite executar uma instrução de cada vez manualmente. Para tal é preciso:

1.1) O robô deve estar em modo manual (Seletor em “Manu”), não existir alarme ativo (sinalizador de alarme “F” a verde) e estar todas as drives habilitada (sinalizador de estado 15 no menu Controlo a verde) conforme a figura abaixo.



1.2) Selecionar a instrução a executar, clicar no botão “Executar” de seguida aparecerá uma mensagem para confirmação conforme a figura abaixo.

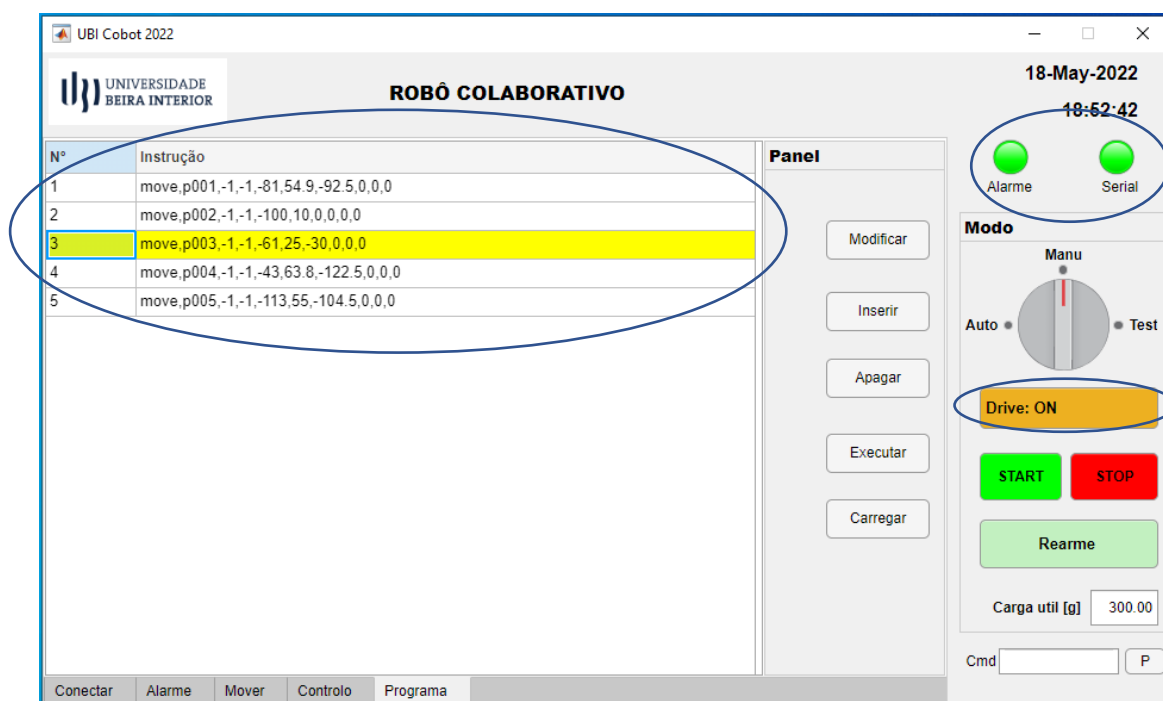


1.3) Confirmar a execução clicando em “SIM”, o robô alcançará a posição automaticamente.

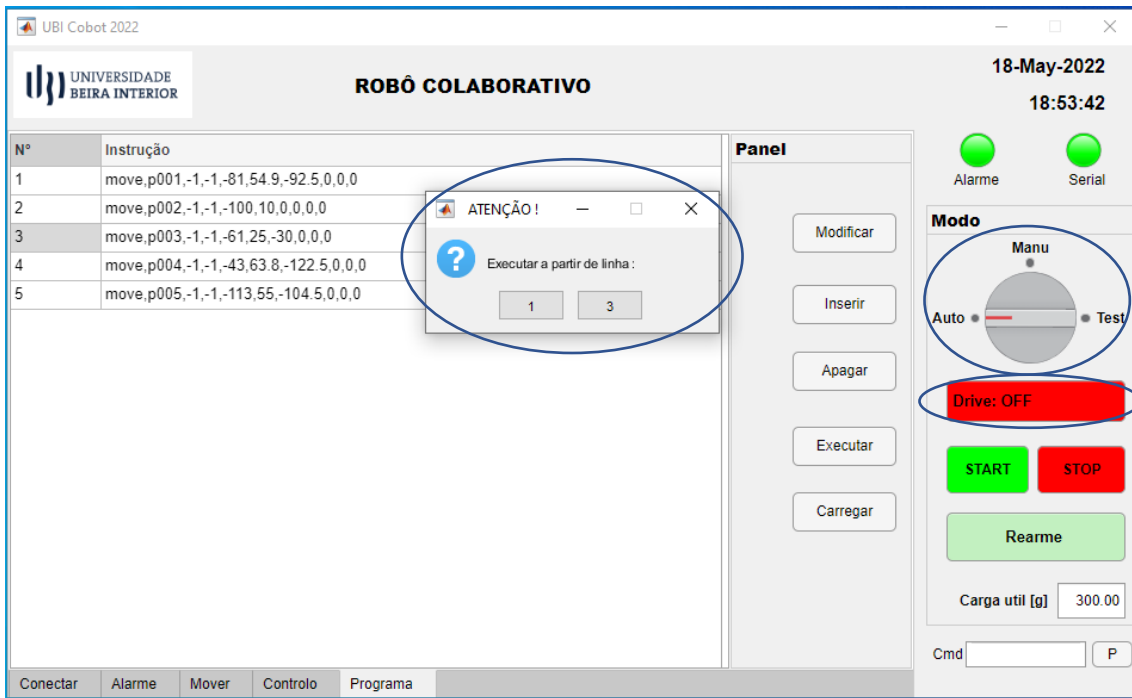
Executar instrução em automático

Este procedimento permite executar as instruções uma de cada vez automaticamente num ciclo repetitivo. Para tal é preciso:

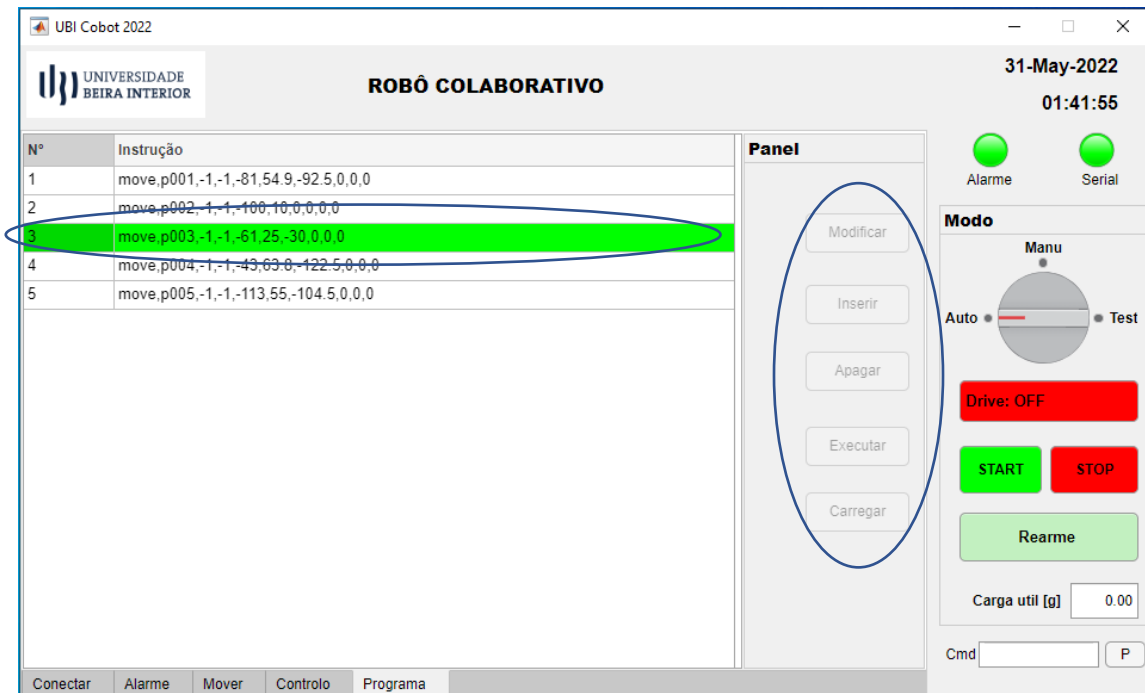
1.1) Não deve existir alarme ativo (sinalizador de alarme “F” a verde), ter todas as drives habilitada (sinalizador de estado 15 no menu Controlo a verde) e selecionar a linha ao qual pretende iniciar a execução conforme a figura abaixo.



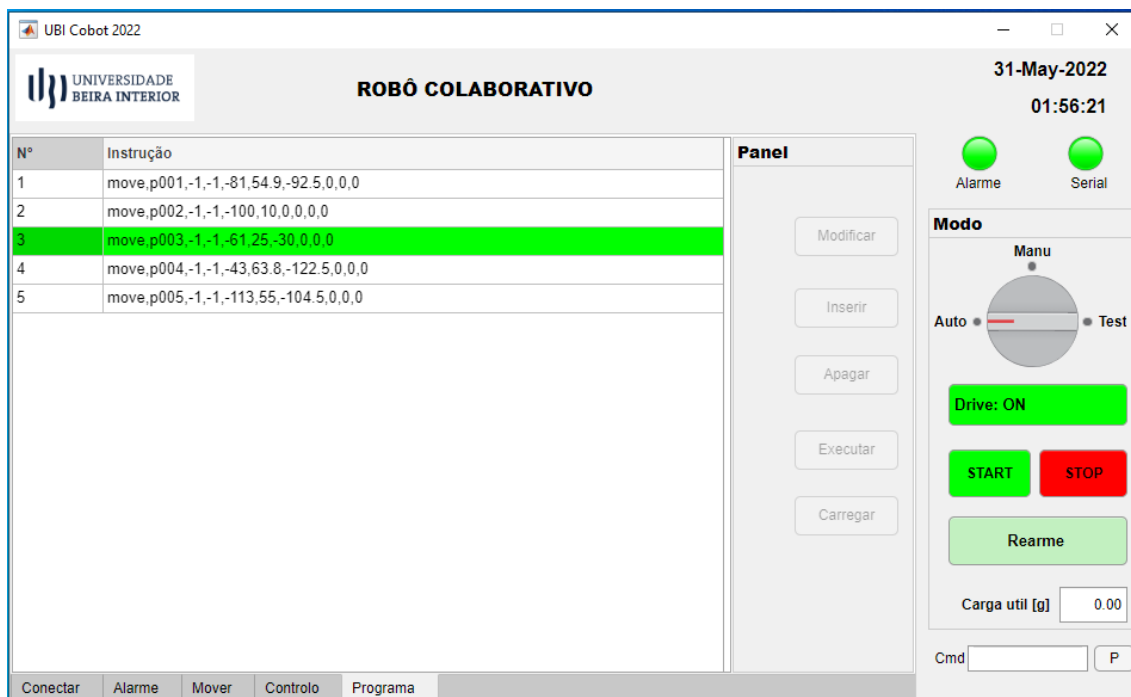
1.2) Virar o seletor para modo automático “Auto”, a drive por segurança desabilitará automaticamente e posteriormente aparecerá uma mensagem para selecionar de aonde pretende inicializar a execução (do início ou a linha selecionada) conforme a figura abaixo.



1.3) Após selecionar a linha ao qual inicializará a execução, a linha selecionada passar a verde e alguns botões será bloqueado automaticamente conforme a figura abaixo.



1.4) Clicar no botão “START”, a drive habilitará e o robô entrar em execução conforme a figura abaixo.



1.5) A execução pode ser interrompida caso surgir alarme, o seletor for mudado para outro modo ou caso for clicado no botão “STOP”. Para retomar a execução, se for clicado no botão “STOP”, será necessário voltar a clicar no botão “START” caso contrário deve rearmar o defeito e proceder os tópicos anterior (1.1, 1.2, 1.3 e 1.4).

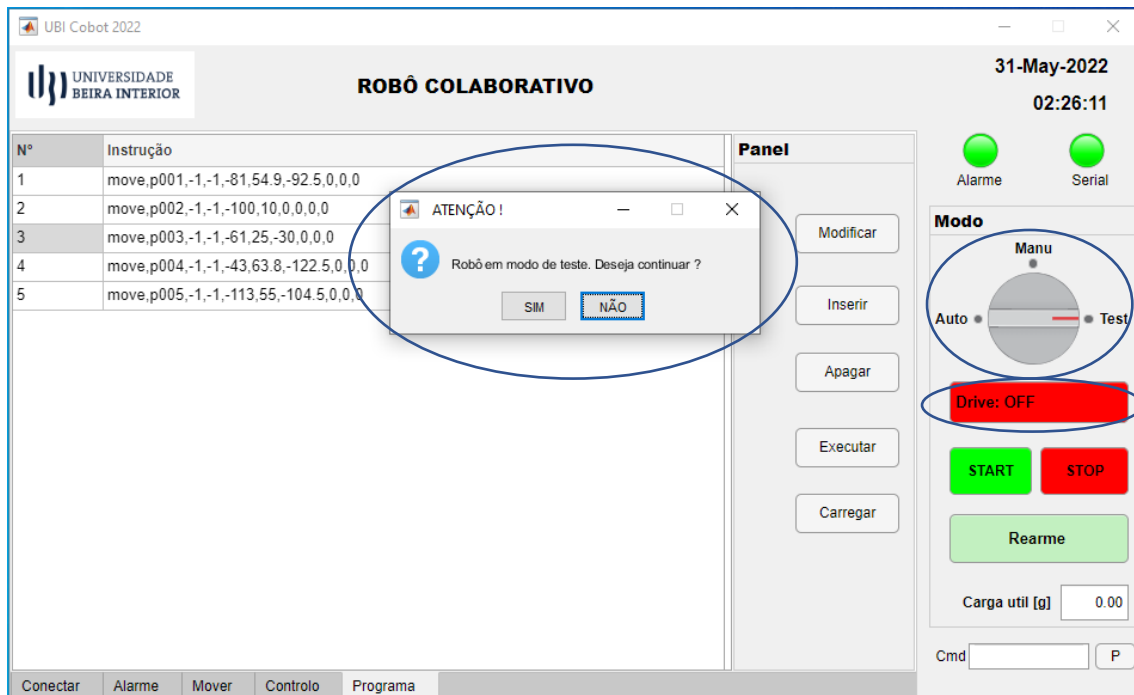
Nota: Durante a execução é possível navegar por outros menus, sendo que algum botão estará bloqueado, mas é possível alterar a percentagem da velocidade e o tipo de passo. Este modo de execução pode ser selecionado a partir de qualquer menu, caso o modo automático for selecionado fora de menu Programa, o menu Programa será selecionado e carrega o programa automaticamente e será selecionado a primeira linha caso existe automaticamente.

Execução em modo teste

Neste modo de execução, o robô irá executar uma sequência de ponto aleatoriamente com a finalidade para registar picos de corrente elétrica consoante a carga útil, erro entre a posição atual e desejada e são guardados automaticamente num ficheiro de texto posteriormente para ser analisados. Para o tal deve:

1.1) Não deve existir alarme ativo (sinalizador de alarme “F” a verde), ter todas as drives habilitada (sinalizador de estado 15 no menu Controlo a verde).

1.2) Virar o seletor para o modo “Test”, a drive por segurança desabilitará automaticamente e posteriormente deve confirmar a execução conforme a figura abaixo.



1.3) Confirmar a execução clicando no botão “SIM”, posteriormente clicar no botão “START” para habilitar drive e o robô entrará em execução.

Nota: Este modo de execução pode ser selecionado a partir de qualquer menu.

Gestão de alarme

Os alarmes estão codificados e numerados de 1 a 140, sendo que foram reservados 20 código de defeito para cada ESP32 (ESP principal, ESP eixo 1, eixo 2, eixo 3, eixo 4, eixo 5 e eixo 6) sendo que dos 20 códigos, existe algumas reservas para trabalho futuro. A tabela abaixo apresenta o código de alarme, a respetiva descrição, causa e solução.

Código	Descrição	Causa	Solução
1	Falha comunicação com ESP32 de eixo1	Excedeu o tempo de resposta (parâmetro 21 de menu controlo)	a
2	Falha comunicação com ESP32 de eixo2		
3	Falha comunicação com ESP32 de eixo3		
4	Falha comunicação com ESP32 de eixo4		
5	Falha comunicação com ESP32 de eixo5		
6	Falha comunicação com ESP32 de eixo6		
20	Defeito rotação do motor de eixo 1	Falta de variação de valor do ângulo durante a execução	b
21	Possível colisão no eixo 1	Aumento de corrente elétrica, motor em esforço	c
22	Necessita de calibração eixo 1	ESP32 inicializou, necessidade de referenciar o encoder	d
23	Movimento não autorizado no eixo 1	Variação do ângulo > 0.1 ° sem ordem de execução	e
24	Eixo 1 atingiu o limite	Robô atingiu o seu limite predefinido	f
25	Falha de comunicação com driver (TIC500) eixo 1	Todos os dados recebidos nulos.	g
26	Defeito driver (TIC 500) de eixo 1	ESP32 recebeu erro de TIC500	h
27	Baixa tensão de alimentação no TIC500 de eixo 1	Tensão de alimentação inferior a 24 Volts	i
28	TIC500 de eixo 1 em defeito (Serial Error)	ESP32 recebeu erro de TIC500	h
29	Timeout driver do eixo 1	ESP32 recebeu erro de Timeout	j
30	Erro sentido movimento eixo 1	Aumento de erro durante a execução	k
31	Drive do eixo 1 não habilitado	Detetou drive desabilitada durante a execução	l
32	Ângulo não alcançável no eixo 1	Aumento de erro durante aproximação	m
33	Falha de comunicação com ADS1115 do eixo 1	Erro na comunicação I2C entre ESP32 e ADS1115	n
40 ... 53	Erro igual ao do eixo 1, apenas com a descrição para eixo 2		
60 ...73	Erro igual ao do eixo 1, apenas com a descrição para eixo 3		
80 ... 93	Erro igual ao do eixo 1, apenas com a descrição para eixo 4		
100 ...113	Erro igual ao do eixo 1, apenas com a descrição para eixo 5		
120 ...133	Erro igual ao do eixo 1, apenas com a descrição para eixo 6		

Legenda das letras:

a	Desligar o ESP32 e volta a ligar ou verificar correspondência no endereço MAC
b	Verificar ligação entre motor e drive, encoder e ESP32 ou ESP32 e drive ou prisão no eixo do motor
c	Obstáculo no percurso ou veio do motor preso
d	Efetuar calibração
e	Verificar existência de movimento no eixo do motor ou a ligação de encoder e ESP32
f	Diminuir/aumentar a posição alcançar
g	Verificar as ligações e alimentação na drive.
h	Consultar o manual de TIC500
i	Verificar a fonte de alimentação
h	TIC500 em defeito, consultar o manual
j	Verificar a ligação entre TIC500 (RX) e ESP32(TX)
k	Verificar a ligação de sentido do motor ou de encoder
l	Verificar a ligação entre TIC500 e ESP32 (TX e RX ou RX e TX)
m	Diminuir a resolução da posição alcançar
n	Verificar a ligação entre ADC e ESP32 (SCL, SDA) ou alimentação (5 V)

Conclusão geral

Quando a comunicação não for estabelecida entre interface de utilizador - PC e ESP32 principal, quase todos os botão e comando na interface ficarão bloqueados enquanto não se estabelecer a comunicação.

Durante a execução, se ocorrer alarme num dos eixos, automaticamente todos os ESP32 interrompem a execução. Qualquer anomalia, o(s) ESP32(s) entra(m) em segurança sinalizando o erro detetado.

Se houver a necessidade de substituir um dos ESP32, é necessário definir o endereço MAC do mesmo no ESP32 principal e no programa do mesmo, contendo a referência do eixo de acordo com a estrutura abaixo.

Para ESP32 principal:

```
24 uint8_t Address_master[] = {0x58, 0xBF, 0x25, 0x9A, 0x42, 0x54};
25 uint8_t broadcastAddress[3][6] = { {0x78, 0xE3, 0x6D, 0x09, 0xF8, 0x54}, //MAC esp32 Eixo 1
26                                     {0x78, 0xE3, 0x6D, 0x09, 0xE3, 0x9C}, //MAC esp32 Eixo 2
27                                     {0x24, 0x0A, 0xC4, 0xEE, 0xE0, 0xB0} //MAC esp32 Eixo 3
28                                     }; //End MAC'S
```

Para ESP32 dos eixos:

```
48 uint8_t Address_master[]={0x58, 0xBF, 0x25, 0x9A, 0x42, 0x54};
49 String String_mact[] = {"78:E3:6D:09:F8:54", "78:E3:6D:09:E3:9C","24:0A:C4:EE:E0:B0"};
```

O programa desenvolvido (código) no IDE do Arduino para o ESP32 principal é diferente para os ESP32 dos eixos. O código para ESP32 dos eixos são iguais apenas são diferenciados dos eixos através do endereço MAC. Tem de haver uma correspondência entre endereço MAC devido o protocolo ESP NOW e também no processo automático na identificação do eixo.