10 AUTOMAÇÃO, ROBÓTICA E CONTROLO INDUSTRIAL – INDÚSTRIA 4.0

ROBÓTICA INDUSTRIAL PARTE I

INTRODUÇÃO, PROGRAMAÇÃO BÁSICA E MANUTENÇÃO

FILIPE PEREIRA
JOSÉ MACHADO

engebook

AUTORES
FILIPE PEREIRA
JOSÉ MACHADO

TÍTULO
ROBÓTICA INDUSTRIAL
PARTE I - INTRODUÇÃO, PROGRAMAÇÃO BÁSICA E MANUTENÇÃO

COLEÇÃO
AUTOMAÇÃO, ROBÓTICA E CONTROLO INDUSTRIAL – INDÚSTRIA 4.0

EDIÇÃO
Quântica Editora – Conteúdos Especializados, Lda.
Praça da Corujeira n.º 38 · 4300-144 PORTO
Tel: 220 939 053 · E-mail: geral@quanticaeditora.pt · www.quanticaeditora.pt

CHANCELA
Engebook – Conteúdos de Engenharia

DISTRIBUIÇÃO
Booki – Conteúdos Especializados
Tel. 220 104 872 . Fax 220 104 871 . E-mail: info@booki.pt . www.booki.pt

PARCEIRO DE COMUNICAÇÃO
Robótica - Revista Técnico-científica de Automação, Controlo e Instrumentação - www.robotica.pt

REVISÃO
Quântica Editora – Conteúdos Especializados, Lda.

DESIGN
Delineatura – Design de Comunicação · www.delineatura.pt

APOIO
Roboplan, Lda. - www.roboplan.pt

IMPRESSÃO
Abril, 2023

DEPÓSITO LEGAL
495493/22

A **cópia ilegal** viola os direitos dos autores.
Os prejudicados somos todos nós.

Copyright © 2023 | Todos os direitos reservados à Quântica Editora – Conteúdos Especializados, Lda.
A reprodução desta obra, no todo ou em parte, por fotocópia ou qualquer outro meio, seja eletrónico, mecânico ou outros, sem prévia autorização escrita do Editor e do Autor, é ilícita e passível de procedimento judicial contra o infrator.

Este livro encontra-se em conformidade com o novo Acordo Ortográfico de 1990, respeitando as suas indicações genéricas e assumindo algumas opções especificas.

CDU
681.2 Instrumentação
681.5 Engenharia de Controlo Automático. Tecnologia Inteligente.

ISBN
Papel: 9789899101081
E-book: 9789899101098

Catalogação da publicação
Família: Automação Industrial
Subfamília: Automação Industrial

roboplan.pt

ÍNDICE

AGRADECIMENTOS E DEDICATÓRIAS XI
NOTA INTRODUTÓRIA XIII

1. INTRODUÇÃO À ROBÓTICA INDUSTRIAL 15
1.1. Definição de *robot* 17
1.2. *Robot*s industriais, AGV's e robótica móvel 17
1.3. Leis da robótica 18
1.4. Comparação *robot* – ser humano 18
1.5. Constituição de um *robot* 20
1.6. Elos e juntas num *robot* 22
1.7. Graus de liberdade 23
1.8. Área de trabalho de um *robot* 24
1.9. Precisão de movimentos 25
1.10. Tipos básicos de juntas ou articulações 26
1.11. Notação dos movimentos do punho 27
1.12. Classificação de *robot*s ao nível configuração física 28
1.12.1. *Robot*s cartesiano ou retangular 29
1.12.2. *Robot*s cilíndricos 30
1.12.3. *Robot*s esféricos 31
1.12.4. *Robot*s articulados ou antropomórficos 32
1.12.5. *Robot*s SCARA 33
1.12.6. *Robot*s Delta (aranha) 34
1.13. Classificação de *robot*s ao nível do tipo de Sistemas de Controlo 34
1.13.1. Controlo Ponto a Ponto - PTP 35
1.13.2. Trajetória Contínua - CP 35
1.13.3. Trajetória Controlada 36
1.13.4. Controlo Inteligente 36
1.14. Tipos de acionamentos em *robot*s industriais 36
1.15. Como escolher/selecionar um *robot* industrial 37
1.16. Tipos de garras para *robot*s industriais 38
1.17. Tipos de ferramentas para *robot*s industriais 40

1.18. Tipos de programação de *robots* industriais 40
1.19. Classificação dos métodos de programação 42
1.20. Segurança na robótica 44
1.21. Aplicações industriais de *robots* 46
1.22. Como mover várias juntas num *robot* industrial? Cálculos cinemáticos 48
1.23. Células e linhas montagem com *robots* industriais 49
1.24 Segurança 50

2. CINEMÁTICA E POSICIONAMENTO DE MANIPULADORES ROBÓTICOS 53
2.1. Indicações gerais para trabalho com segurança 57
2.1.1. Solução ou cinemática direta 57
2.1.2. Solução ou cinemática inversa 59
2.2. Resolução trigonométrica das equações que posicionam a garra de um micro *robot* TeachMover 59
2.2.1. Solução ou cinemática direta 59
2.2.2. Solução ou cinemática inversa 61
2.3. Transformações homogéneas 64
2.3.1. Translação 65
2.3.2. Rotação 66
2.4. Interpretação de transformações homogéneas 66
2.5. Solução direta do TeachMover pelo método matricial 67
2.5.1. Definição dos sistemas de referência no TeachMover 68
2.5.2. Definição das matrizes 69
2.5.3. Equações cinemáticas do TeachMover 71
2.5.4. Obter a solução inversa de forma matricial 73

3. CONTROLADOR YRC1000 77
3.1. *Hardware* do Controlador YRC1000 79
3.2. Ligando o Controlador YRC1000 80
3.3. Desligando o Controlador YRC1000 82

4. CONSOLA DE PROGRAMAÇÃO OU *TEACH PENDANT* YRC1000 83
4.1. Interruptor de seleção de modos de funcionamento 87
4.1.1. Modo *Teach*; Ícone de *Status* 87
4.1.2. Modo *Play*; Ícone de *Status* 87
4.1.3. Modo Remoto [*Play*]; Ícone de *Status* 87
4.2. Botões de operação 87
4.2.1. Botão de Emergência (*E-STOP*) 87
4.2.2. Botão de *Start* 88
4.2.3. Botão *Hold*; Ícone de *Status* 88
4.2.4. Botão Servo *On/Ready* 88
4.3. Botão para ligar Servos (*Deadman*) 88

4.4. Botão *Area* ... 89
4.5. Tecla *Cursor* ... 89
4.6. Tecla *Select* ... 90
4.7. Tecla *Cancel* ... 90
4.8. Teclas de edição ... 90
4.8.1. Teclado numérico ... 90
4.8.2. Tecla *Enter* ... 91
4.8.3. Tecla *Insert* ... 91
4.8.4. Tecla *Modify* ... 91
4.8.5. Tecla *Delete* ... 91
4.9. Teclas de Programação ... 91
4.9.1. Tecla *Motion Type* ... 91
4.9.2. Tecla *Inform List* ... 92
4.10. Teclas *Shift* ... 92
4.11. Tecla *Interlock* ... 92
4.12. Tecla *Main Menu* ... 92

5. INTERFACE DO YRC1000 ... 93
5.1. Menu principal ... 95
5.2. Ecrã principal ... 96
5.3. Área de menus ... 97
5.4. Área de interface humana ... 98
5.4.1. Linha de edição de *Buffer* ... 98
5.4.2. Linha de mensagem ... 98
5.5. *Display* de *Status* ... 99
5.6. Opções do menu principal ... 100
5.6.1. *Job* (Programa) ... 101
5.6.2. *Arc Welding, General, Handling, Spot Welding, Laser, Paint*, etc. ... 101
5.6.3. *Variable* (Variáveis) ... 101
5.6.4. *In/Out* (Entradas/Saídas) ... 101
5.6.5. *Robot* ... 102
5.6.6. *System Info* (Informações de Sistema) ... 102
5.6.7. Ex. *Memory* ... 102
5.6.8. *Setup* (Configurações) ... 102
5.6.9. *Display Setup* (Configurações de Tela) ... 102
5.7. Ecrãs de introdução de dados ... 105

6. GRUPOS, SISTEMAS DE COORDENADAS E EIXOS ... 107
6.1. Grupos de controlo ... 109
6.1.1. Tecla *Robot* ... 109
6.1.2. Tecla Ex. AXIS ... 109
6.2. Sistemas de coordenadas ... 110

6.2.1. Coordenada *Joint* ... 111
6.2.2. *Robots* de 7 Eixos [SLURBT e E] ... 112
6.2.3. Seleção de ferramenta para movimentação em coordenadas XYZ ... 112
6.2.4. Coordenadas Retangular, Cilíndrica, *Tool* e *User* ... 113
6.2.5. Coordenada *Tool* ... 116
6.2.6. *User Frames* ... 117
6.3. Ecrãs e ícones de velocidade manual ... 117
6.3.1. Velocidade manual ... 118
6.3.2. Tecla *High Speed* (Alta velocidade) ... 118
6.4. Posição atual ... 119
6.5. Posições de programa ... 119
6.6. Exercício 1 ... 121

7. *SECOND HOME POSITION* ... 123

8. CRIAÇÃO DE PROGRAMAS EM LINGUAGEM *INFORM* ... 129
8.1. Criação de novo programa ... 131
8.2. Aceder ao programa ativo ... 132
8.3. Selecionar um programa ... 133
8.4. *Master Job* ... 134
8.4.1. Registar o nome de um *Master Job* ... 134
8.4.2. Exibindo o *Master Job* ... 136
8.5. Exercício 2 ... 137

9. CRIAÇÃO DE TRAJETÓRIAS ... 139
9.1. Posição gravada ... 141
9.2. Tipos de movimento ... 141
9.3. Velocidade ... 142
9.3.1. Seleção de valores de velocidade da lista ... 143
9.3.2. Seleção manual de velocidades ... 143
9.4. Programar uma linha de movimento ... 143
9.5. Movimento *Joint* ... 145
9.6. Movimento Linear ... 146
9.7. Movimento Circular (MOVC) ... 147
9.8. Movimento Spline (MOVS) ... 149
9.9. Trajetórias com pontos iguais no início e fim de trajetórias ... 150
9.10. Exemplo de sobreposição do primeiro e último passo ... 151
9.11. Exercício 3 ... 152
9.12. Exercício 4 ... 153

10. CONFIRMAÇÃO DE TRAJETÓRIA ... 155
10.1. Seleção de ciclo ... 158

10.2. Execução do programa ... 160
10.3. Exercício 5 ... 160
10.4. Exercício 6 ... 161

11. COPIAR, APAGAR E RENOMEAR PROGRAMAS ... 163
11.1. Copiar programas ... 165
11.2. Apagar programas ... 166
11.2.1. Habilitando a função de recuperar programas ... 166
11.2.2. Apagar programas ... 166
11.2.3. Como recuperar programas apagados ... 167
11.3. Renomear programas ... 167
11.4. *Job Header* ... 168
11.5. Ordenar programas por nome/data ... 169
11.5.1. Como ordenar a lista de programas por nome ou data ... 169
11.5.2. Janela de detalhes ... 169
11.6. Exercício 7 ... 170

12. EDIÇÃO DE PROGRAMAS ... 171
12.1. Inserir linhas no programa ... 174
12.2. Modificar uma posição já previamente gravada ... 175
12.3. Apagar uma posição já previamente gravada ... 175
12.4. Modificação do tipo de movimento ... 176
12.5. Resumo ... 176
12.6. Copiar, cortar, colar ... 177
12.6.1. Função *Copy* (Copiar) ... 177
12.6.2. *Cut* (Cortar) ... 178
12.6.3. *Paste* (Colar) ... 179
12.6.4. *Reverse Paste* (Colar Inverso) ... 180
12.7. Edição geral de uma linha de programa ... 181
12.8. *Detail* (Edição de Detalhes) ... 182
12.9. Velocidade de rotação ou angular ... 183
12.10. Nível de posicionamento ou *Position Level* ... 184
12.11. *Positioning Level Tag* ... 185
12.12. *Corner Radius Tag* ... 187
12.13. *Tag FPT* ... 188
12.14. COORD ... 189
12.15. Linhas com comentários ... 190
12.16. Exercício 8 ... 191
12.17. Exercício 9 ... 191
12.18. Exercício 10 ... 191

13. EDIÇÃO DE VELOCIDADE 193
13.1. Lista de velocidades 195
13.2. Entrada de dados 196
13.3. Edição de detalhes 196
13.4. *Change Speed* 197
13.5. *Speed* (Velocidade) 199
13.6. *Cycle Time* (Janela de Tempo de Ciclo) 199

14. PROTEÇÃO CONTRA EDIÇÃO EM PROGRAMAS 201
14.1. *Edit Lock*: proteção de programa 203
14.2. Alteração de movimentos para programas com *Edit Lock* 203
14.3. Exercício 11 204

15. ALARMES E ERROS 205
15.1. Mensagens de erro 207
15.2. Alarmes menores 207
15.3. Alarmes maiores 208
15.4. Ecrã de alarme 209
15.5. Histórico de alarmes 210
15.6. *Overrun* e *Shock Sensor* 211
15.7. *Shock Sensor* interno e deteção de colisão 212
15.8. Alarmes múltiplos 215

16. MANUTENÇÃO 217
16.1. Frequência de inspeção e manutenção 219
16.2. Inspeções diárias 220
16.3. Inspeções cíclicas em função do nº de horas de trabalho 220
16.4. Calibração do *robot* 221
16.5. Análise de alarmes e erros 221
16.6. Procedimentos de *backup* 223

17. EXERCÍCIOS EXTRA 227
17.1. Exercício de trajectória linear e circular 229
17.2. Exercício de trajectória linear e circular – Desenho automóvel 229

REFERÊNCIAS BIBLIOGRÁFICAS CCXXXI
ÍNDICE DE FIGURAS CCXXXIII

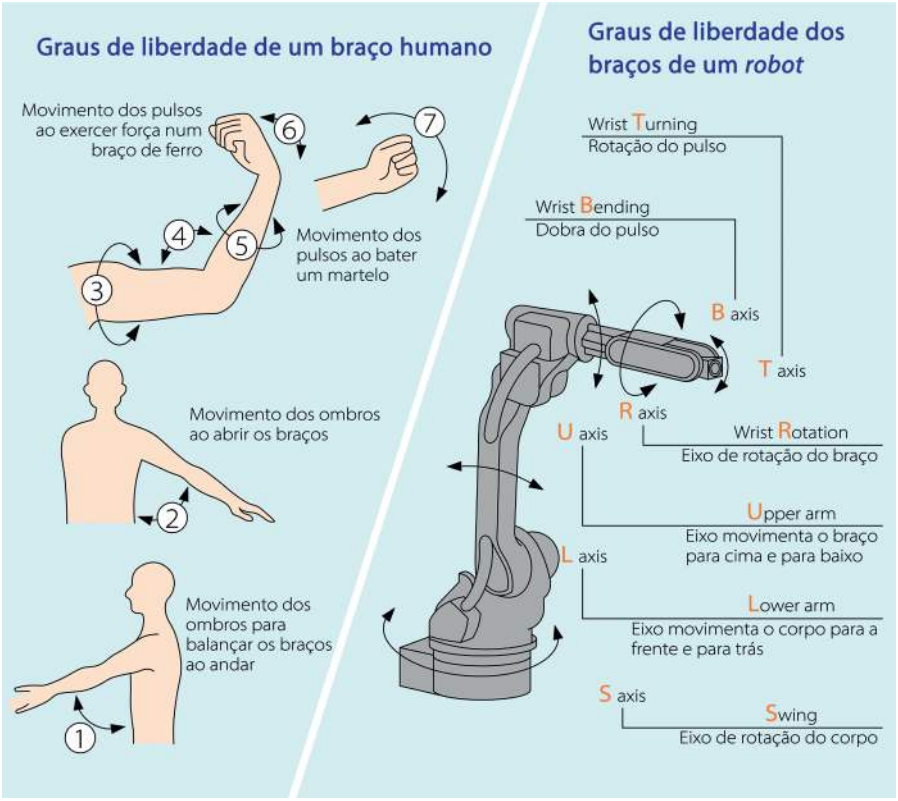


Figura 1.3. *Correspondência Ser Humano/Robot da YASKAWA*

1.5. Constituição de um *robot*

Um *robot* é constituído basicamente por uma base fixa, um braço articulado composto de elementos ou (elos) mecânicos, um controlador, *teach pendant* ou consola e sensores. Os manipuladores podem ser reprogramados, para além de que recebem informações externas de sensores de realimentação, permitindo o correto posicionamento de peças ou ferramentas colocadas na extremidade dos mesmos na flange ou punho.

Em resumo, os componentes fundamentais que constituem um *robot* industrial são:

- O BRAÇO MECÂNICO;
- OS SENSORES;
- O PROCESSAMENTO E CONTROLO (Controlador);
- REDE DE COMUNICAÇÕES (de série ou opcional);
- ALIMENTAÇÃO;
- DIÁLOGO HOMEM – MÁQUINA ou *TEACH PENDANT* ou CONSOLA.

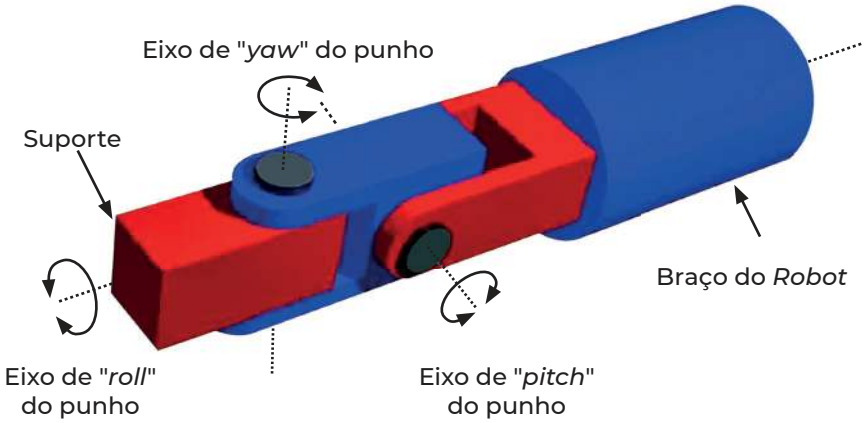


Figura 1.14. *Tipos de movimentos do punho num robot industrial*

1.12. Classificação de *Robots* ao nível configuração física

Normalmente os *robots* são classificados de acordo com as suas configurações físicas ou de acordo como sistema de controlo adotado.

Ao nível da classificação baseada na configuração física, estes podem ser:

- Cartesiano ou retangulares;
- Cilíndrico;
- Polar ou esféricos;
- Articulados;
- SCARA.

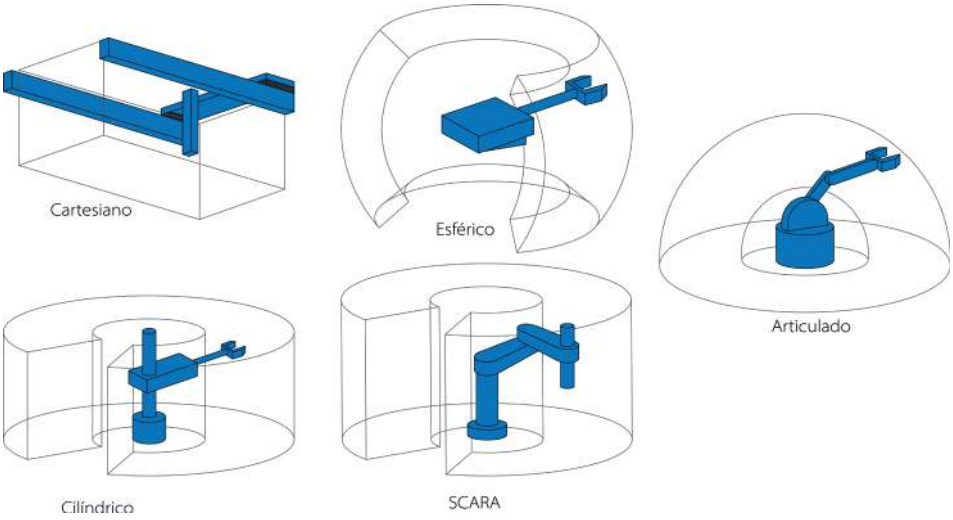


Figura 1.15. *Classificação dos robots baseada na configuração física*
Fonte: Adaptado de https://slideplayer.com.br/slide/10605215/

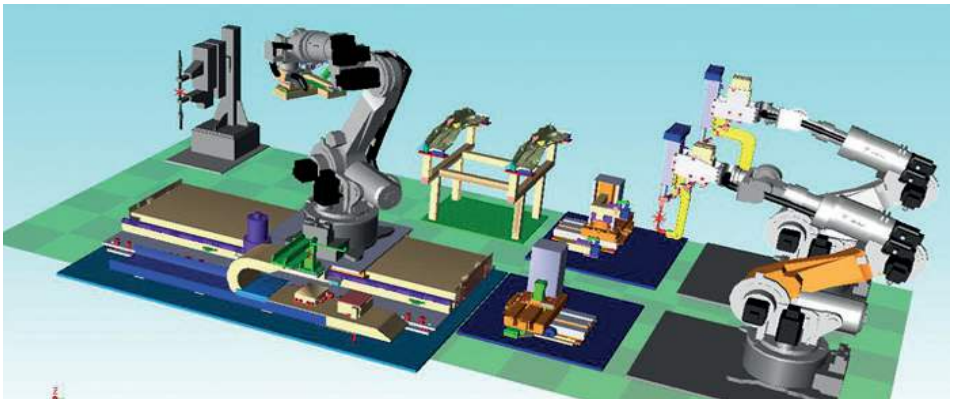


Figura 1.28. *Exemplo de uma simulação offline de um sistema industrial no software MOTOSIM EG-VRC da YASKAWA*

1.19. Classificação dos métodos de programação

Existem dois tipos de Programação de *robots* industriais. A primeira é a chamada programação por aprendizagem (*on-line*) que consiste em ensinar o *robot* guiando-o através da trajetória definida pelo utilizador.

A segunda forma de programação de *robots* industriais é a programação textual (*off-line*), que permite indicar a tarefa ao *robot* mediante o uso de uma linguagem de programação de alto nível.

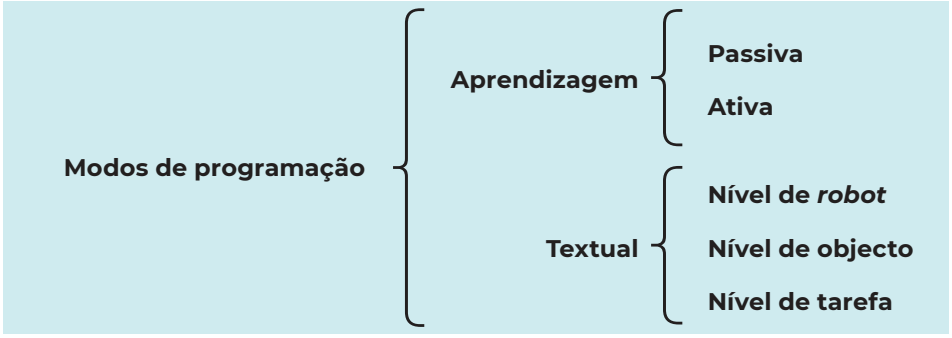


Figura 1.29. *Tipos de programação de um robot industrial*
Fonte: PMR2560 Elementos de Robótica
Departamento de Engenharia Mecatrônica Escola politécnica da USP

No método de programação por aprendizagem, esta pode ser do tipo passiva ou ativa. No caso da passiva, o programador segura o extremo do braço do *robot* e movimenta-o através da trajetória mais adequada. Um exemplo de aplicação deste tipo de programação é na pintura.

No caso da programação por aprendizagem do tipo Ativa, O programador usa uma consola ou *teach-in box* para controlar os motores das articulações ao longo da trajetória. Um

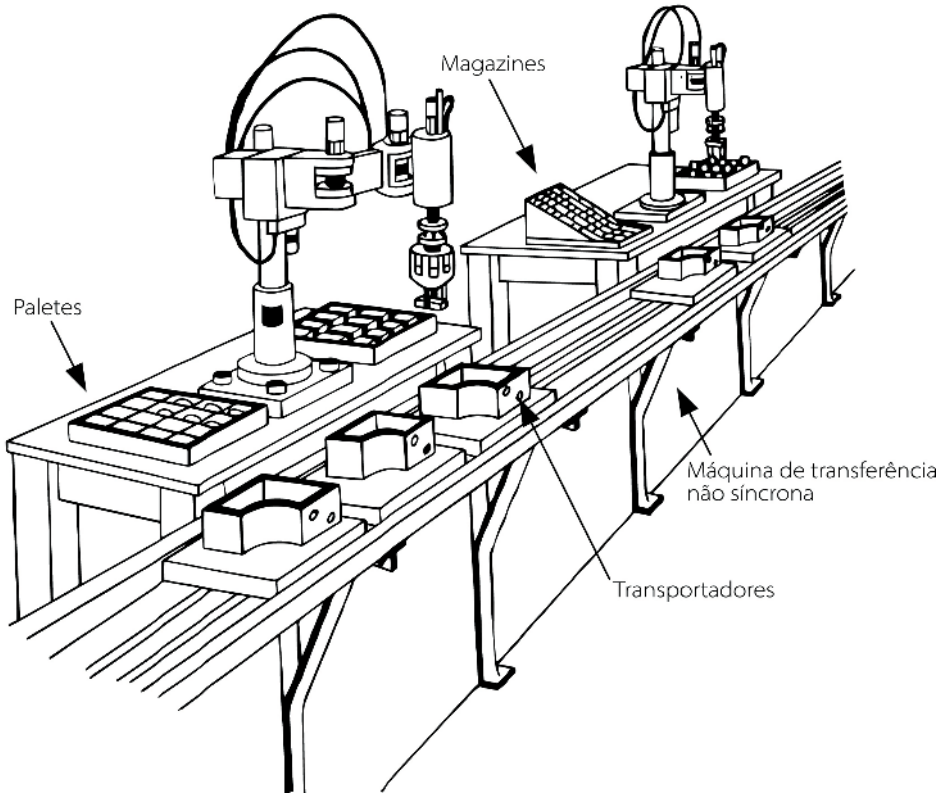


Figura 1.37. *Linha de montagem*

1.24. Segurança

Este capítulo destina-se a descrever os principais procedimentos de segurança que se deve ter em conta ao utilizar *robots* industriais. Não trataremos de como realizar ou instalar um sistema de segurança de uma célula *robotizada*, que deve ser concebida na altura do projeto da célula, pois este tema será abordado num outro volume da coleção.

De referir que a referência que realizamos à segurança de trabalho com o *robot* não dispensa a leitura do capítulo relacionado com segurança do próprio manual dos equipamentos.

Antes de trabalhar com o *robot*, deverá rever os manuais dos equipamentos. Durante a utilização destes equipamentos, irão aparecer frequentemente os seguintes símbolos:

Sinaliza uma situação de elevado perigo, que se não for evitada pode causar a morte ou ferimentos graves.

Sinaliza uma situação de risco médio, que se não for evitada pode causar a morte ou ferimentos graves.

2.1.2. Solução ou cinemática inversa

A solução inversa tenta estabelecer o valor dos parâmetros , H e V, conhecendo a posição do ponto terminal A, por meio das coordenadas (x, y, z).

Na figura 2.4. verifica-se que:

$$H = \sqrt{(x2 + y2)}$$

$$\varphi = \text{arctg } y/x$$

Na figura 2.3. verificamos que:

$$V = L - z$$

2.2. Resolução trigonométrica das equações que posicionam a garra de um micro *robot* TeachMover

A Figura 2.5. mostra o modelo cinemático do manipulador do *robot* de formação TeachMover.

As coordenadas angulares indicadas no modelo cinemático são F1, F2 e F3, além de F4 e F5, que correspondem aos ângulos de rotação da articulação diferencial do punho e cuja composição dá origem aos movimentos denominados PITCH e ROLL. Nos símbolos cinemáticos representados na Fig. 2.5., a direção dos ângulos positivos nas juntas é indicada.

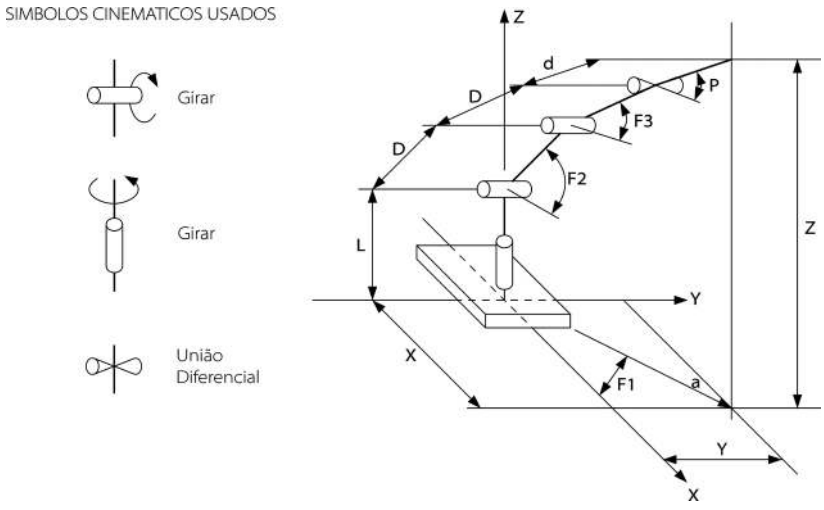


Figura 2.5. *Modelo cinemático do robot TeachMover*

2.2.1. Solução ou cinemática direta

A cinemática direta determina o valor das coordenadas (x, y, z) e o valor dos ângulos PITCH e ROLL com base nas coordenadas da junta, ou seja, os ângulos F1, F2, F3, F4 e F5.

Botão de Paragem de Emergência (*E-STOP*) no Controlador YRC1000

Pressionando o botão de emergência (*E-STOP*) corta a energia dos motores e aciona os freios.

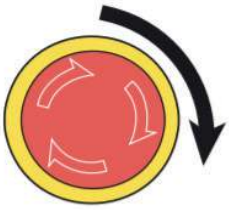


Figura 3.3. *Botão de emergência*

A consola ou o *pendant* mostra a mensagem seguinte.

Robot stops by P. Panel emergency stop

Rode o botão de emergência no sentido horário para soltá-lo.
Todas as rotinas de emergência funcionam da mesma maneira em *Teach*, *Play* ou *Remote*, incluindo qualquer emergência externa.

Robot stops by external emergency stop

NOTA: Os motores não podem ficar alimentados quando o sistema não está em funcionamento. Pressione algum botão de emergência ou mude para o modo de Teach. (Todos os procedimentos de manutenção de graxa do manipulador são baseados nas horas de Servo On).

3.2. Ligando o Controlador YRC1000

A porta ou as tampas do controlador devem estar completamente fechadas e trancadas antes de alimentar o controlador. O controlador inicia o *software* e inicializa as rotinas de diagnóstico do YRC1000.

Numa célula robótica onde o YRC1000 comunica com outros equipamentos, o controlador deve sempre ser mantido com alimentação, de modo que a energia será controlada por outro dispositivo.

Durante o diagnóstico, um ecrã de inicialização é apresentado até que o controlador esteja pronto para operação.

4.1. Interruptor de seleção de modos de funcionamento

O estado de um sistema robotizado é feito através da seleção de 3 diferentes modos (*Teach*, *Play* e *Remote* [*Play*]).

4.1.1. Modo *Teach*; Ícone de *Status*

No modo *Teach*, aparece na barra o seguinte ícone de status onde o utilizador pode movimentar tanto *robots* como eixos externos de forma manual. Neste modo, os menus de programação, edição e personalização ficam disponíveis, baseados no nível de segurança.

4.1.2. Modo *Play*; Ícone de Status

No modo *Play*, aparece na barra o seguinte ícone de status e o botão de *START* torna-se ativo e o controlador YRC1000 tem o Controlo da execução de programas. Modos especiais de reprodução de programas ficam disponíveis.

NOTA: Mudar do modo Play para Teach durante a execução de um programa poderá causar a interrupção da execução.

Para continuar, selecione PLAY, clique em "SERVO ON" e depois START.

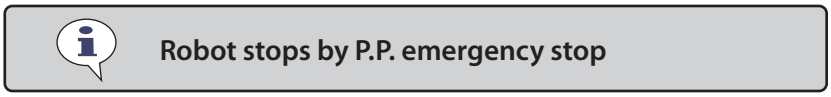
4.1.3. Modo Remoto [*Play*]; Ícone de Status

O modo Remoto permite o controlo do *robot* através de um comando externo opcional, um autómato ou um computador.

4.2. Botões de Operação

4.2.1. Botão de Emergência (*E-STOP*)

Ao pressionar o botão de emergência (*E-STOP*) aparece na barra o seguinte ícone de status , e os servomotores ficam sem alimentação e os freios são acionados.



Aparece no ecrã, e o indicador

apaga.

Essa mensagem tem prioridade sobre qualquer outra mensagem.
Rode o botão de emergência totalmente no sentido horário para que a mensagem desapareça.
Com todos os botões

de emergência não acionados, a Linha de Status mostra o ícone .

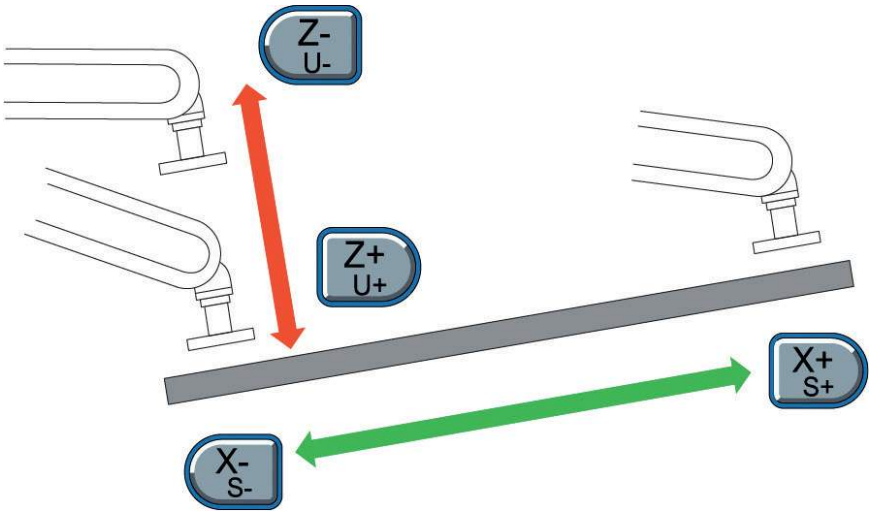


Figura 6.10. *Sentidos de Movimentação em Coordenada TOOL*

A origem de qualquer *TOOL FRAME* sempre é o TCP. Por isso, é impossível que a posição atual do *robot* seja apresentada em função do *TOOL FRAME*.

6.2.6. *User Frames*

O sistema de coordenadas de utilizador (*USER FRAME*) é baseado em pontos criados pelo próprio utilizador, fundamentado no plano de trabalho do *robot*. Podem ser criados até 63 coordenadas de utilizador diferentes.

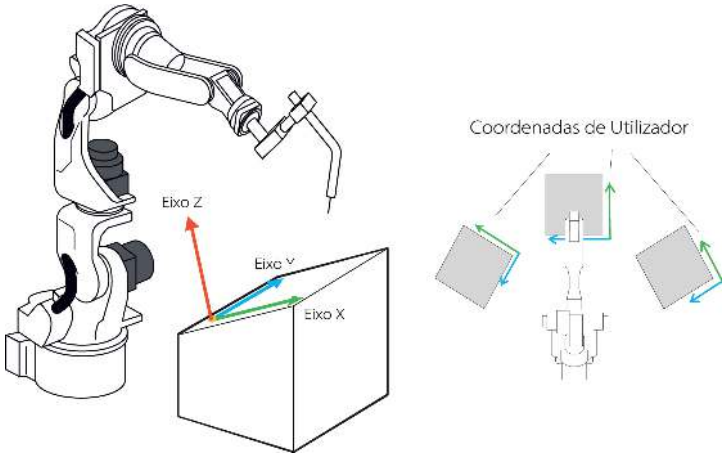


Figura 6.11. *Exemplo de Coordenadas de Utilizador*

6.3. Ecrãs e Ícones de Velocidade Manual

A velocidade manual determina a velocidade que o *robot* ou eixo externo movimentar-se-á com as teclas de movimento ou com FWD/BWD.

NOTA: O robot deve ser movimentado para a posição desejada antes da edição da linha de buffer (linha temporária). Entretanto, se o operador não mantiver os motores alimentados durante a edição do buffer, a posição pode mudar ligeiramente devido à ação dos freios. Uma nova movimentação pode ser necessária para voltar à posição exata desejada antes de adicionar a linha ao programa.

9.5. Movimento *Joint*

Ao selecionar um movimento tipo *JOINT* (articulação), o *robot* utiliza o caminho mais fácil para alcançar a posição programada; deve-se ter em conta que essa trajetória nunca é uma reta, já que o *robot* determina a maneira mais fácil de alcançar o ponto programado. Este tipo de movimento *JOINT* é usado principalmente para aproximar ou afastar o *robot* de uma peça de trabalho, uma vez que neste tipo de movimentos requer-se a velocidade máxima do *robot*; geralmente o primeiro e último passo da trajetória do *robot* são programadas com movimentos tipo *JOINT* (MOVJ).

Os movimentos *JOINT* são inseridos no programa como MOVJ, selecionando um movimento deste tipo, as opções de velocidade variam de 0,01% a 100%, pelo que então deve-se definir a velocidade adequada de acordo com os requisitos do processo.

A velocidade *VJ=Joint_speed* representa a velocidade máxima de cada motor do *robot* durante a movimentação para a posição gravada. (Em alguns movimentos alguns motores podem não se mover, de acordo com a necessidade).

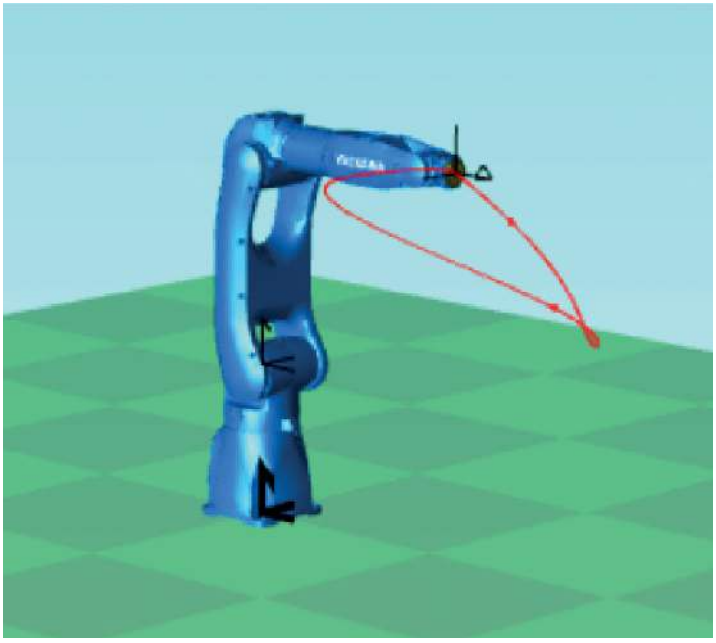


Figura 9.4. *Movimentos Aéreos com Movimento Joint*

10. CONFIRMAÇÃO DE TRAJETÓRIA

Existem dois métodos para rever o caminho programado.

■ Método 1: Teclas Forward FWD e Backward BWD

Comece com o cursor no primeiro passo do programa e verifique se existe um caminho livre para que o robot possa alcançar a posição programada a partir da posição atual.

Selecione uma velocidade manual, de preferência a velocidade média (M). Para mover o *robot* para a etapa em que o cursor está, mantenha pressionada a tecla FWD e, em seguida, o *robot* move-se no determinado passo e para automaticamente quando alcança a posição. Ao mesmo tempo a direção do passo para de piscar. Para mover o *robot* através de cada passo sucessivo, deixe de pressionar a tecla FWD e volte a pressioná-la novamente. Ao pressionar a tecla BWD o *robot* move-se para o passo anterior, de acordo com a posição do cursor.

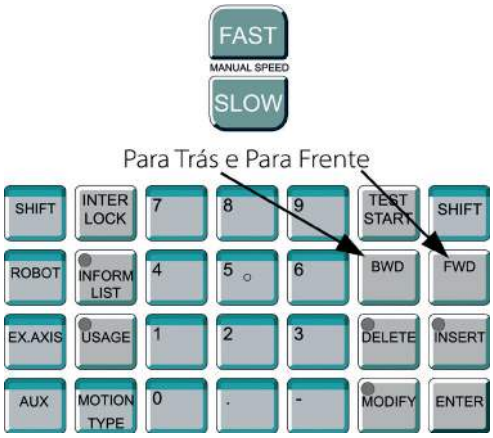


Figura 10.1. *Confirmação de Posição e Trajetória*

O movimento do *robot* é executado apenas enquanto as teclas FWD ou BWD são pressionadas; ao deixar de pressionar qualquer uma das teclas, interrompe-se o movimento do *robot*. A tecla HIGH SPD (*High Speed*) funciona apenas com o FWD, mas não com o BWD.

NOTA: Este procedimento só funciona com instruções de movimentação. Pule as linhas com instruções sem movimentação ou use INTERLOCK+FWD com os motores ligados para executar qualquer função desejada.

CUIDADO! Não selecione outras linhas de movimentação com o cursor durante a execução da trajetória, pois assim o movimento será executado isoladamente pelo controlador, sem interpolações.

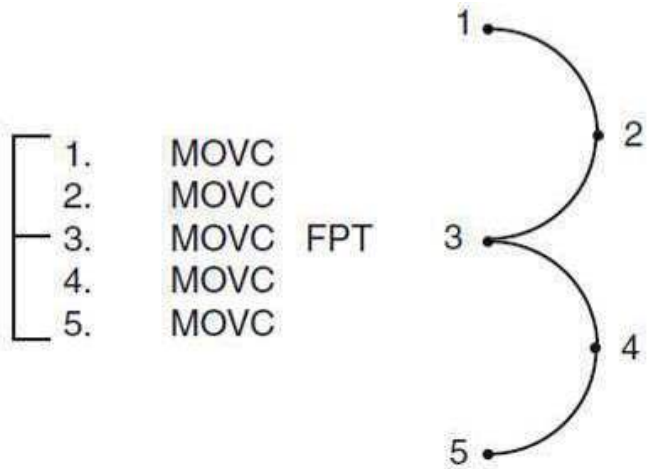


Figura 12.22. *FPT MOVC Consecutivos com o Tag FPT*

12.14 COORD

Como o YRC1000 recalcula a trajetória do TCP ao utilizar MOVCs, a orientação da ferramenta durante a trajetória pode não ser adequada.

A tela de detalhes do MOVC possui a função "Circ Pose Ctrl" com a *Tag COORD*.
Esta *tag* permite o controlo do ângulo de ferramenta que vai de 0 a 1, conforme necessidade de orientação de ferramenta durante a trajetória.

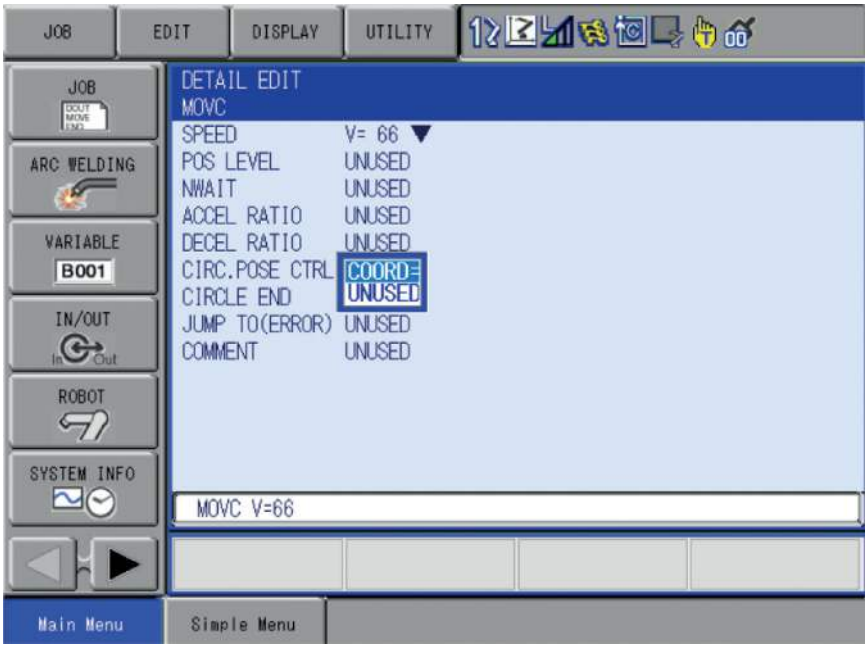


Figura 12.23. *Janela de Detalhe da tag COORD*

Sempre que ocorre um alarme, o manipulador para de imediato e a janela de alarmes surge na consola de programação indicando que o *robot* parou devido a alarme e indicando o nº de alarme como se vê na figura seguinte.

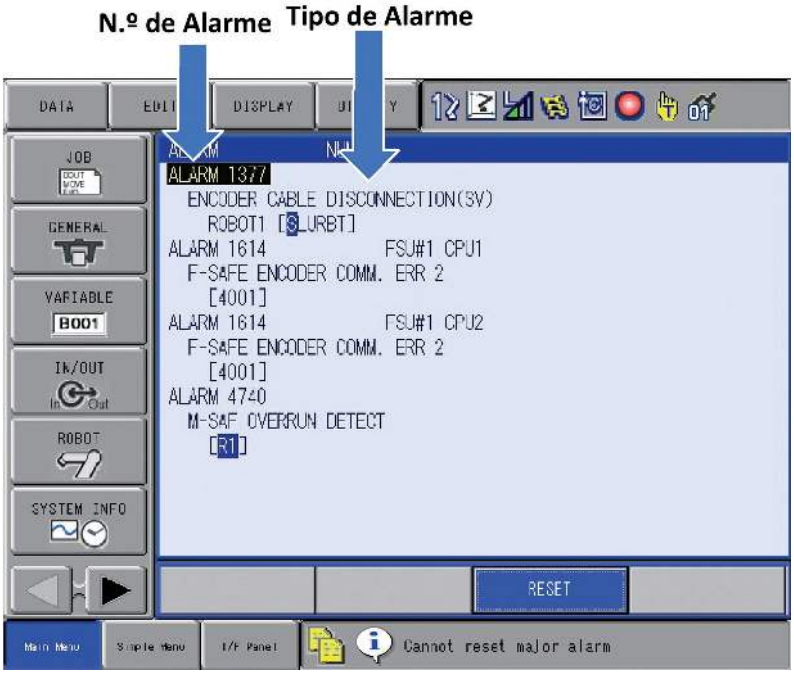


Figura 16.2. *Exemplo de página com alarmes*

Caso haja mais do que um alarme associado, para ver os erros todos tem que se percorrer toda a área de imagem da consola de forma a surgir o novo erro.

A maioria dos alarmes podem ser resolvidas pelo utilizador. Nos casos onde se procede à resolução do alarme segundo o indicado no manual e mesmo assim o alarme mantém-se, então nesse caso deverá contactar a assistência técnica do fabricante do *robot*.

16.6. Procedimentos de *backup*

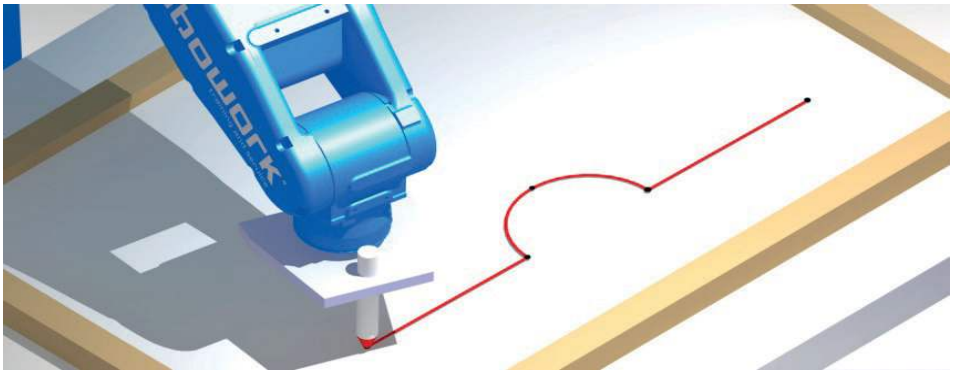
As consolas permitem a ligação de dispositivos de memória externos como *pendrive* USB e, em controladores mais antigos, cartões "*Compact Flash*" (CF). Desta forma é possível guardar programas e parâmetros do *robot* e ter uma cópia desses dados guardados de forma segura.

No caso de haver alguma alteração indesejada desses dados no *robot*, haverá sempre uma cópia que pode ser gravada novamente no *robot*.

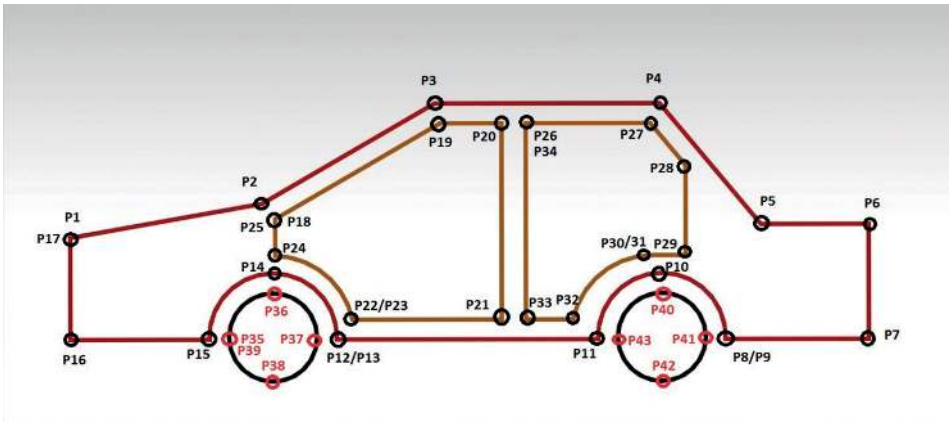
Os passos a realizar para a realização de cópias de segurança são os seguintes, conforme as imagens apresentadas.

17. EXERCÍCIOS EXTRA

17.1. Exercício de trajectória linear e circular



17.2. Exercício de trajectória linear e circular – Desenho automóvel



APOIO

roboplan
robotics experts

+351 234 943 900
info@roboplan.pt
Aveiro
www.roboplan.pt

ROBÓTICA INDUSTRIAL
PARTE I – INTRODUÇÃO, PROGRAMAÇÃO BÁSICA E MANUTENÇÃO

FILIPE PEREIRA
JOSÉ MACHADO

Sobre a coleção

Esta coleção, para além de suprimir uma necessidade ao nível de obras na área da automação, robótica e controlo industrial, dando ênfase à Indústria 4.0 e à digitalização, visa preparar profissionais capazes de conceber e implementar processos de robotização e automatização industrial, promovendo ao longo de todos os volumes a capacidade de adquirir *know-how* para concretizar soluções de digitalização de sistemas e processos, fundamentais para as indústrias do futuro se tornarem mais autónomas e competitivas.

Sobre a obra

A Robótica Industrial é uma área em crescimento, assistindo-se à sua gradual disseminação e lecionação em escolas e instituições de ensino superior. Neste contexto, há porém uma lacuna a nível de bibliografia em língua portuguesa nesta área, tanto para quem ensina, para alunos do ensino superior e do ensino profissional/técnico, para engenheiros e técnicos responsáveis por esta área, como ainda para o público em geral. Este livro resulta sobretudo da necessidade de preencher essa omissão no domínio da formação em Robótica Industrial, tendo os autores utilizado a marca de *robots* YASKAWA neste volume não só por ser uma das marcas mais vendidas em todo o mundo, mas também pela sua fácil operação e programação.

Sobre os autores

Filipe Pereira é mestre e especialista em Eletrónica, Automação, Robótica e Controlo Industrial e possui 22 anos de experiência profissional na área. É atualmente membro investigador efetivo no CIETI - Centro de Inovação em Engenharia e Tecnologia Industrial e membro colaborador do Centro de Investigação MEtRICs da Universidade do Minho.

José Machado doutorou-se em Engenharia Mecânica - Automação, pela Universidade do Minho (Portugal) e em simultâneo, pela École Normale Supérieure de Cachan (França) em 2006. Obteve o título de Agregado, na Universidade do Minho, em 2019. É Diretor Adjunto do Centro de Investigação MEtRICs e Professor Auxiliar com Agregação no Departamento de Engenharia Mecânica da Universidade do Minho. É autor ou coautor de mais de 250 publicações em revistas e conferências internacionais indexadas e coordena(ou) vários Projetos de Investigação e Transferência de Tecnologia nos domínios da Mecatrónica e Automação.

Apoio

roboplan robotics experts

robótica

Também disponível em formato *e-book*

ISBN: 978-98-99-101-08-1

www.**engebook**.pt

engebook