

# UM CONTROLADOR DE ARQUITETURA ABERTA APLICADO A UM ROBÔ REIS RV-15 PARA AVALIAÇÃO DE PROCESSOS DE MEDIÇÃO E SOLDAGEM ROBOTIZADA

EMERSON P. RAPOSO, MARCELO R. STEMMER, RAFAEL M. TAOUIL

*Departamento de Automação e Sistemas (DAS), Centro Tecnológico (CTC)  
Universidade Federal de Santa Catarina (UFSC)*

*Caixa Postal:476, CEP: 88040-900 Florianópolis, SC, BRASIL*

*E-mails: er@das.ufsc.br, marcelo@das.ufsc.br, maytaouil@das.ufsc.br*

**Abstract**— This work demonstrates the benefits that an open architecture controller (OAC) for robots provides in the execution of measurement and welding tasks assisted by a robot. A new controller has been conceived for an existing industrial robot, providing access to its hardware and software components and consequently allowing new studies such as: implementation of classic or neural control strategies; modeling DH-kinematics or screws; integration of different effectors such as plasma welding devices or laser sensors for measurement; execution of trajectories with linear, quadratic or splines interpolation, etc. Finally, the advantages of using an open architecture controller for the robot are presented for a measurement and a welding process.

**Keywords**— Open Architecture Controller (OAC), Open Robot Control, Industrial Robots.

**Resumo**— Este trabalho demonstra os benefícios que um controlador aberto para robôs proporciona na execução de medição e soldagem auxiliadas por um robô. A partir de um robô industrial pode-se conceber um novo controlador, este aberto, permitindo acesso a seus componentes sendo *hardware* e *software*, e conseqüentemente, viabilizando aplicações de novos estudos tais como: estratégias de controle clássicas ou neurais; modelagem cinemática DH ou helicóides; diferentes efetores de soldagem Plasma ou sensores folha de luz para medição; execução de trajetórias com interpolação linear, quadrática ou *splines*. Por fim, as vantagens de usar um controlador de arquitetura aberta para robô são apresentadas para os processos de medição e soldagem.

**Palavras-chave**— Controlador de Arquitetura Aberta, Controle de Robô Aberto, Robôs Industriais.

## 1 Introdução

Um robô é composto de seu manipulador e controlador. Atualmente, os avanços às pesquisas e aplicações demandam cada vez mais acesso aos componentes que fazem parte do controlador (FORD, 1994) (LIMA, 2001) (PRITSCHOW, 2005) (KOREN, 2005). Assim, a motivação deste trabalho é apresentar os resultados da aplicação de um controlador de arquitetura aberta (OAC), suas vantagens, para avaliação de processos de medição e soldagem utilizando um robô denominado REIS.

Este artigo brevemente comentou a demanda por OAC. Na seção 2 serão apresentados o manipulador e controlador do robô, destacando suas respectivas características técnicas que serviram de suporte ao projeto do OAC. Em seguida, na seção 3 serão descritos os problemas que justificam o desenvolvimento e implementação de um OAC com o objetivo de realizar medição e soldagem de uma cratera usinada em chapa de aço para simular o efeito da erosão por cavitação, envolvendo aspectos tais como: estratégia de controle, modelo cinemático, integração de efetores e execução de trajetória. O controlador de arquitetura aberta é explanado, apresentado a sua concepção do ponto de vista *hardware* e *software* no decorrer da seção 4. Por fim, seção 5, os resultados obtidos em função da aplicação de um OAC para avaliação dos processos de medição e soldagem robotizada serão relatados.

## 2 Descrição do Robô

O controlador para robôs é formado pelo servo-amplificador de potência e pelo seu sistema computacional segundo a ISO (*International Standards Organization*) norma 8373:1994. Assim, quanto mais flexível for o controlador de um robô aumenta-se a chances de pesquisa, desenvolvimento e inovação tecnológica. E, conseqüentemente, os benefícios da robótica em áreas até então pouco exploradas atualmente (RAPOSO, 1998) (ADÔRNO, 2005) (PAIM, 2005).

A seguir serão apresentados o elemento de manipulação e o controlador utilizado neste trabalho destacando suas principais características.

### 2.1 Manipulador REIS

O robô industrial REIS foi montado no Brasil pela empresa IPSO Automatização Ltda, hoje extinta, com peças trazidas da Alemanha. Ele é um robô com seis graus de liberdade, sendo estes originados por 6 juntas rotativas. O acionamento é elétrico, através de motores CC sem escovas. Para a medição da posição angular das juntas dispõe-se de um *encoder* ótico incremental montados em seus respectivos atuadores. Este manipulador é típico das aplicações da indústria. Ele é composto de um braço antropomórfico (juntas 1, 2 e 3) e um punho esférico (juntas 4,5 e 6), como mostra a figura 1.

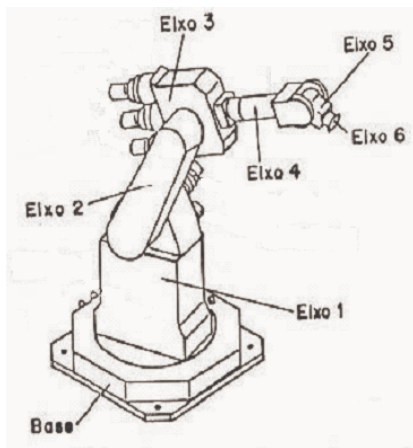


Figura 1: Manipulador REIS

Estão descritas abaixo as principais características técnicas do robô:

- Fabricante: (<http://www.reisrobotics.com>)
- Modelo: RV-15
- Carga útil = 15 Kg, com baricentro a 133 mm do ponto de giro do eixo 5.
- Espaço de trabalho: figura 2 (Dimensões em milímetros).

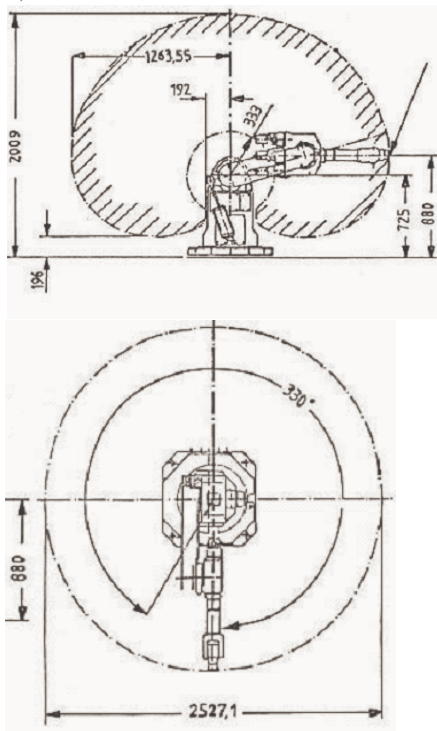


Figura 2: Vista lateral e superior do espaço de trabalho.

No REIS, os *encoders* geram 1000 pulsos altos a cada giro completo do motor (ou seja, têm 1000 ranhuras), o que significa, na prática, 4000 pulsos de contagem da posição angular. Como existe uma redução de 100 vezes entre o motor e a junta, um giro do *encoder* corresponde a 3,6 graus de giro na respectiva junta do robô. Assim, o giro completo de uma junta do robô equivale a 400000 pulsos. Estas relações valem para todos os motores do manipulador.

Devido a limitações mecânicas, as juntas têm fins de curso. A tabela 1 mostra a excursão completa de cada uma das juntas do manipulador REIS. Estes valores não são os mesmos que os do fabricante. A razão disto é que o zero dos *encoders* foi reconfigurado. As juntas 4 e 6 não têm limites mecânicos, os limites impostos são apenas lógicos, ou seja, implementados em *software*.

Tabela 1: Excursão total de cada junta.

Junta	Excursão (graus)	Excursão (pulsos)
1	$0^\circ \leq \theta_1 \leq 319^\circ$	$0p \leq \theta_1 \leq 354.000p$
2	$0^\circ \leq \theta_2 \leq 135^\circ$	$0p \leq \theta_2 \leq 150.000p$
3	$0^\circ \leq \theta_3 \leq 270^\circ$	$0p \leq \theta_3 \leq 300.000p$
4	$-180^\circ \leq \theta_4 \leq 180^\circ$	$-200000p \leq \theta_4 \leq 200.000p$
5	$0^\circ \leq \theta_5 \leq 229^\circ$	$0p \leq \theta_5 \leq 254.000p$
6	$0^\circ \leq \theta_6 \leq 360^\circ$	$0p \leq \theta_6 \leq 400.000p$

## 2.2 Controlador REIS

A figura 3 apresenta o controlador industrial do REIS. Nele encontra-se os servo-amplificadores e seu sistema computacional. O servo-amplificador do REIS é formado por seis placas com amplificadores de tensão.

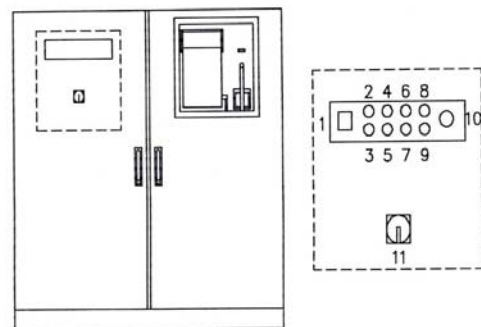


Figura 3: Controlador do REIS

Na tabela 2 são mostradas as características técnicas das placas de potência. Para os eixos principais 1, 2, e 3 é utilizada a placa modelo 615, e para os eixos 4, 5 e 6, o modelo 608. Ambas fabricadas pela *Transaxe*.

Tabela 2: Dados técnicos dos amplificadores.

Modelo	608	615
Corrente <sub>Regime</sub> (A)	8	15
Corrente <sub>max</sub> (A)	16	30
Potência (W)	1920	3600
Tensão (V)	250	250

Em cada placa amplificadora se dispõe de uma estratégia de controle, baseada em PID, que a partir de seus potenciômetros pode-se estabelecer e regular qual melhor esquema em função da aplicação. O sistema computacional é dedicado, utiliza barramento de comunicação VME, processador e sistema operacional dedicados.

### 3 Controlador de Arquitetura Aberta

Desde a década de 80 há tentativas de concepção e aplicação de controladores de arquitetura aberta (ALBUS et al., 1989 1994) (LUMIA, 1989 1994) (MILLER, 1991) (ANDERSON, 1993) (STEWART, 1992 1997) (HISSAM, 2004) (KOREN, 1996 2005). A seguir serão relatados os problemas envolvidos na utilização de um robô pra realizar soldagem Plasma. A partir destes problemas, são apresentados os detalhes que motivaram criar um OAC a fim de avaliar os processos de medição e soldagem plasma auxiliado por um robô aberto.

**Estratégia de controle:** No começo das atividades com o REIS, as trajetórias não eram realizadas de forma suave e existiam ruídos nos motores. Esses movimentos trêmulos não permitiam uma boa qualidade da solda. Após vários testes descobriu-se que seriam necessários ajustes em cada estratégia de controle das respectivas juntas do robô, pois os parâmetros estavam inadequados.

**Modelo cinemático:** Durante testes de validação da cinemática direta e reversa, observou-se que a posição alcançada pelo efetuador era diferente. A diferença principal entre essas configurações, é basicamente o giro dos eixos 4, 5 e 6. Esse giro é grande (180°) nos eixos 4 e 6 e isso gera o erro de posicionamento. O erro é intrínseco à construção mecânica e acontece sempre quando juntas que utilizam sistemas de transmissão (engrenagens ou correia) e que tem o seu acionamento através de motores localizados em elos diferentes daquele onde está fixa a transmissão. A solução é fazer uma compensação da propagação de erros nas respectivas juntas através de seu modelo cinemático.

**Integração de ferramentas:** Com o intuito de incorporar um sistema de medição e soldagem conjuntamente com suas ferramentas e respectivos processos foi necessário obter a matriz de transformação de cada efetuador. A tocha de solda e o sensor de medição foram submetidos à calibração. Com a calibração feita, se conheceram exatamente quais são as translações e rotações desde a interface do manipulador até o bico da tocha de solda. A matriz de transformação homogênea foi incorporada ao *software* e utilizada em conjunto com as cinemáticas. Como as cinemáticas levam em conta apenas os elos do manipulador, é necessário realizar algum tratamento matemático para que se possa utilizar o robô com os seus efetuadores acoplados. O valor correto da posição da ferramenta é obtido através da multiplicação da matriz homogênea resultante da cinemática direta pela matriz proveniente da calibração.

**Executivo tempo real:** A execução de trajetória do robô necessitava ser realizada de duas maneiras e com *deadline* associado a cada junta do manipulador: trajetória ponto-a-ponto e trajetória contínua.

Os movimentos com *blend move* foram a primeira forma de executar trajetórias, ponto-a-ponto, que foi implementada e testada. Foram usadas informações de posição, velocidade e aceleração. Traduzindo-se *blend move* por movimentos mesclados, a técnica consiste em sobrepor dois movimentos da forma trapezoidal. Portanto, caracteriza-se um movimento trapezoidal aquele cuja velocidade em função do tempo forma um trapézio. Os extremos desse trapézio representam a rampa de aceleração e desaceleração. A mesclagem de dois trapézios é feita no momento que ocorre a desaceleração do primeiro movimento. Nesse instante, ocorre a superposição da rampa de desaceleração do primeiro movimento com a rampa de aceleração do segundo movimento. O resultado é uma velocidade que varia de forma contínua e menos brusca que uma rampa de desaceleração seguida de uma rampa de aceleração. Isso permite menos oscilações nos movimentos do robô e por conseqüência uma solda de melhor qualidade. Apesar deste método ter se mostrado bastante confiável nas chapas planas, vários problemas atribuídos à movimentação (trajetória e orientação) ocorreram nos ensaios com chapas cavitadas (côncavas). Nas extremidades dos cordões de solda onde existe mudança de sentido de movimento e nos pontos onde a cavidade é mais acentuada exige uma maior movimentação para manter a orientação correta, conseqüentemente, apareceram falhas de soldagem. Esses problemas foram resolvidos com os movimentos utilizando *buffer*, descritos a seguir.

A segunda forma de executar os movimentos, trajetória contínua, carrega para um *buffer* interno do controlador um vetor com posições em coordenadas de junta relativas a uma trajetória. A execução da trajetória se dá de forma que, a cada intervalo fixo de tempo a junta estará em uma das posições definidas no vetor de posições. Ou seja, são pontos espaçados igualmente no tempo. Esse intervalo pode ser preestabelecido dentro de uma faixa que é relacionada com a frequência de amostragem do controlador PID. Assim, para executar a trajetória dispõe-se de três arquivos especificando as coordenadas de juntas, velocidades e acelerações referente à trajetória desejada.

Os problemas anteriormente expostos não puderam ser resolvidos utilizando o controlador original do robô, pois havia necessidade acrescentar/retirar ou atualizar algoritmos de estratégia de controle, modelo cinemático, integração de ferramentas e executivo tempo real.

Assim, foi projetado um novo controlador, figura 3, um controlador de arquitetura aberta baseado em computador pessoal (*Personal Computer* – PC) utilizando barramento de comunicação PCI e sistema operacional *Windows NT 4.0*, aproveitando-se do elemento de manipulação e amplificador de potência do REIS.

A seguir serão relatados os aspectos do OAC, denominado OPENREIS, subdividido em *hardware* e *software*.



Figura 3: OPENREIS.

### 3.1 Hardware do OPENREIS

A figura 4 representa todas as interligações entre os componentes que formam o sistema necessário para a utilização do robô. Os números entre parênteses representam a quantidade de cada item.

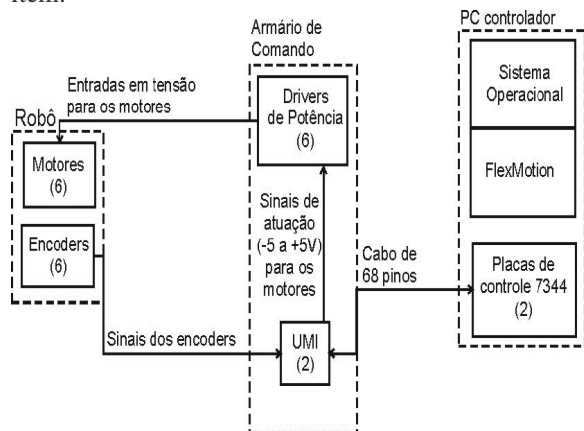


Figura 4: Diagrama de blocos do OPENREIS.

No PC, duas placas de controle de movimento PCI da família 7344 do fabricante *National Instruments* equipam o robô. Cada placa tem capacidade de comandar até quatro servo-motores. Incluem também: quatro portas de entrada/saída de 8 bits, mesclagem de movimentos e memória *FLASH*. O controlador propriamente dito é um PID com velocidade de  $62\mu\text{s}$  por eixo, estimador de velocidade, *feedforward* de posição e velocidade. A interface entre as placas PCI no PC e os *drivers* de potência é feita por placas *National Instruments* UMI-7764 (*Universal Motion Interface*). Como entrada, elas recebem os fios que transmitem os sinais dos *encoders*. Para trocar dados com a placa 7344 existe uma porta de 68 pinos.

### 3.2 Software do OPENREIS

O diagrama de distribuição especificado em UML (*Unified Modeling Language*), figura 5, mostra o modelo da estrutura do OPENREIS, ou seja, de que forma seus elementos estão distribuídos e como interagem entre si. São em três camadas onde cada uma representa um nível de abstração. Considerando a camada de mais baixo nível, próximo ao robô, a camada #1, denominada de camada física. Acima desta, tem-se a camada lógica e, por fim, a camada #3, camada de aplicação.

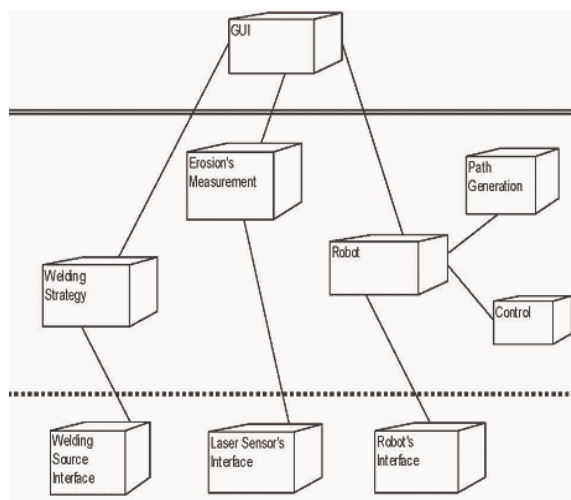


Figura 5: Diagrama de distribuição do OPENREIS.

A seguir, a descrição de cada pacote do OPENREIS:

*GUI*: responsável pela interface com usuário.

*Robot*: pacote que trata dos aspectos dos modelos cinemáticos/dinâmicos do robô.

*Path Generator*: pacote responsável por implementar o gerador de trajetória.

*Control*: pacote que implementa a estratégia de controle do robô.

*Welding Strategy*: pacote que define a melhor estratégia para soldagem.

*Erosion's Measurement*: pacote que tem como fim a medição da região cavitada.

*Weld Source Interface*: biblioteca para a interface com a fonte de solda.

*Laser Sensor's Interface*: biblioteca para a interface com o sensor laser.

*Robot's Interface*: biblioteca para a interface com o manipulador.

O sistema funciona basicamente da seguinte maneira: O operador controla o robô pelo computador. A unidade de potência é a parte intermediária robô-computador e é a responsável pelas fontes de tensão que o alimentam. O sensor laser e a tocha são as ferramentas que o robô utiliza. Sendo que o sensor escaneia a superfície cavitada e a tocha alimentada pela fonte realizam a solda.

## 4 Resultados

Os resultados deste trabalho serão apresentados a seguir, primeiramente, descrevendo a bancada experimental utilizada e os dois processos utilizados: medição e soldagem (TAOUIL, 2004).

### 4.1 Bancada de Ensaio

Após toda concepção do robô aberto, cerca de 40 ensaios de soldagem foram realizados. Eles foram feitos inicialmente com chapas planas e depois evoluíram para chapas fresadas que simulavam uma superfície cavitada. Todos os testes foram feitos com o corpo de prova posicionado com uma inclinação de 45° sobre-cabeça, situação de pior caso para soldagem. A bancada utilizada consiste do OPENREIS incluindo fonte de soldagem microprocessada, módulo plasma, alimentador de arame e suporte à chapa ou corpo de prova, como mostra a figura 6.



Figura 6: Bancada de Experimentação.

### 4.2 Processo de Medição

Tendo sido o robô referenciado, o primeiro passo para se realizar a soldagem é a medição da cavidade. Com o sensor óptico fixado ao OPENREIS, foi realizada a medição inicial da superfície do corpo de prova, figura 7.

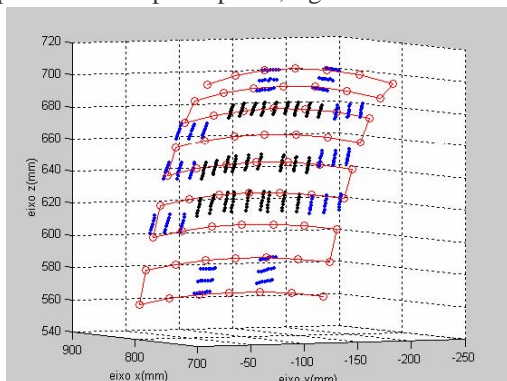


Figura 7: Trajetória de medição automatizada.

Os pontos em azul, pontos externos, definem a região não danificada que envolve a cavitação. Os pontos em preto, pontos internos, caracterizam a forma da região danificada. A trajetória em vermelho do tipo zig-zag representa a medição automatizada da cavidade do corpo de prova.

O resultado do processo de medição pode ser visto na figura 9.

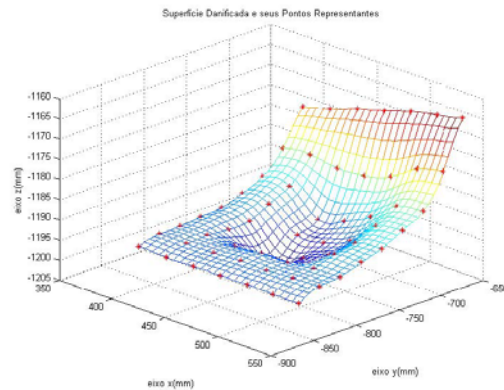


Figura 9: Cratera digitalizada.

### 4.3 Processo de Soldagem

A partir do processo de medição, algoritmos matemáticos e de interpolação reconstróem as superfícies danificada e original. A superfície original é o modelo que se deseja alcançar com o processo de soldagem. A figura 10 mostra, como exemplo, a superfície danificada e a localização dos pontos, em preto, referentes às trajetórias dos cordões de solda da primeira camada de soldagem.

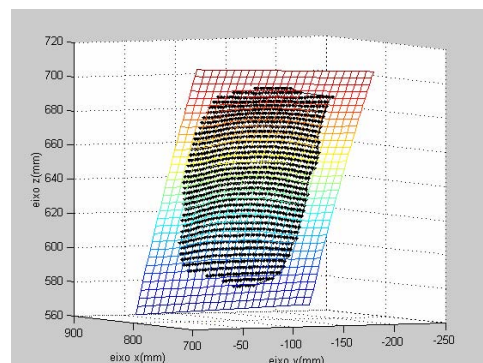


Figura 10: Trajetória de soldagem automatizada.

Com a tocha de soldagem plasma fixado ao OPENREIS, foi realizada a adição de material de acordo com a trajetória indicada pela figura 10 e já convertida para o espaço de juntas, controlando os movimentos e o disparo da fonte de solda juntamente com o alimentador automático de arame.

O Resultado final do processo de soldagem é mostrado na figura 11.



Figura 11: Corpo de prova preenchido com três camadas de solda auxiliado pelo OPENREIS.

#### 4 Conclusão

Graças ao OAC pode-se viabilizar a utilização de um robô para realizar os processos de medição e soldagem integrados. Além disso, a qualidade da soldagem robotizada é superior quando comparado com os processos de deposição de solda manual. Conseqüentemente, esta peça chave para a automação poderá ser cada vez mais popular em nosso cotidiano. E tudo indica que outros setores também poderão absorver os benefícios desta tecnologia. A contribuição da implementação do modelo OPENREIS dá a chance de experimentá-lo em diferentes situações tal que se possa avaliar resultados reais.

#### Agradecimentos

Aos funcionários e professores do departamento de automação e sistemas; do departamento de engenharia elétrica e do departamento de mecânica da UFSC; Ao Instituto de Tecnologia para o Desenvolvimento (LACTEC), Companhia Paranaense de Energia (COPEL), Financiadora de Estudos e Projetos (FINEP), Furnas Centrais Elétricas S. A. (FURNAS), Programa de Capacitação de Recursos Humanos para Atividades Estratégicas (RHAE) e Programa de Apoio ao Desenvolvimento Científico e Tecnológico (PADCT) pelo oportuno desafio e apoio financeiro.

#### Referências Bibliográficas

Adorno, B. V., Aguiar, C. S. R. e Borges, G. A. (2005). Planejamento de Trajetória para Robô OMNI Utilizando o Algoritmo Mapa de Rotas Probabilístico, *Simpósio Brasileiro de Automação Inteligente (SBAI)*.

- Albus, J.S., Lumia, R., Fiala, J., e Wavering, A. (1989). NASREM - The NASA/NBS Standard Reference Model for Telerobot Control System Architecture, *Proc. of the 20th International Symposium on Industrial Robots*, Tokyo, Japan.
- Anderson, R. J. (1993). SMART: A Modular Architecture for Robotic and Teleoperation, *Proc. of the IEEE Int. Conf. on Robot. And Autom.*, Atlanta, Georgia.
- Ford, W. E. (1994). What is an Open Architecture Robot Controller?, *IEEE*.
- Hissam, A. S. e Klein, M. (2004). A Model Problem for an Open Robotics Controller, <http://www.sei.cmu.edu/publications/documents/04.reports/04tn030.html>
- Koren, Y., Pasek, Z.J., Ulsoy A.G. e Wright, P.K., (1996). Timing and Performance of Open Architecture Controllers, *Proceedings of the ASME Dynamics Systems and Control Division*, DSC-Vol.58.
- Koren, Y. (2001). What is a Reconfigurable Manufacturing Systems (RMS)?; *CIRP*.
- Lima, C.R.E, Rosário, J. M., Dumur, Didier (2001). Open Architecture Design for Mobile Robotics. *2nd IFAC Workshop on Intelligent Assembly and Disassembly*.
- Lumia, R. (1989); NASREM: a functional architecture for control of the flight telerobotic servicer, *Journal of Second European In-Orbit Operations Technology Symposium*.
- Lumia, R. (1994). Using NASREM for telerobot control system development, *Robotica*.
- Miller, D.J. e Lennox, R.C. (1991). An Object-oriented Environment for Robot System Architectures. *Control Systems Magazine, IEEE* Vol. 11, Issue 2, Feb., Page(s):14 – 23.
- Paim, P. K., Jouvencel, B. e Borges, G. A. (2005). Controle Reativo para o Robô Submarino TAIPAN; *SBAI*.
- Pritschow, G. (2005). Self-Adapting Control Systems for RMS, *3a. International Conference on Reconfigurable Manufacturing*, May 10-12, Ann Arbor, MI, USA.
- Raposo, P. E. e Stemmer, M. R. (1998) Um Sistema de Reconhecimento de Fala Aplicado a Interação com um Robô. *XII Congresso Brasileiro de Automática*, Uberlândia, MG.
- Stewart, D.B., Volpe, R.A. e Khosla, P.K. (1997). Design of dynamically reconfigurable real-time software using port-based objects, *Software Engineering, IEEE*, Vol. 23, Issue 12, Dec., Page(s):759 – 776.
- Taouil, R. M., Raposo, E. P. e Bonacorso, N. G. (2004). Ensaio, Desenvolvimento e Implantação de Software pra Soldagem Robotizada. *DAS 5501: Relatório de estágio em Controle e Automação*, Florianópolis, SC.