

PLATAFORMA DE PROGRAMAÇÃO VISUAL DEDICADO ROBÔ CARTESIANO RD46

Paolo Raphael Silva de Albuquerque¹, José Henrique d'Souza²

Centro Federal de Educação Tecnológica do RN – CEFET/RN

Departamento Acadêmico de Tecnologia Industrial – DATIN

Núcleo de Desenvolvimento em Mecatrônica – NUDEM

Av. Salgado Filho, 1159 Tirol CEP 59.015-000 Natal-RN

E-mail: paoloraphael@natal.digi.com.br¹, ricky@cefetrn.br²

RESUMO

O presente trabalho apresenta um software para o sistema RD46 da Didacta Itália existente no Laboratório de Robótica do CEFET-RN. O ambiente de programação é visual e está sendo desenvolvido em linguagem de programação Java, possibilitando sua execução em qualquer plataforma (Windows, Linux e Solaris) que possua a Máquina Virtual Java instalada. O objetivo deste sistema é substituir completamente o programador original, o qual é executado sob o MS-DOS e é pouco intuitivo para o operador. O novo sistema prevê a execução passo a passo e seu monitoramento adequado. O software permite que programas sejam gravados e lidos posteriormente para realização de tarefas mais trabalhosas. O novo sistema possibilitará aos alunos do CEFET utilizarem o RD46 nas aulas de Robótica e em atividades de pesquisa no tocante à programação de robôs.

PALAVRAS-CHAVE: sistema de programação; Java; programação visual, programação de robôs.

1. INTRODUÇÃO

O software de controle do robô RD46 apresenta uma interface simples formada por caixas de texto, botões e comboboxes com o objetivo de tornar a programação do manipulador uma tarefa simples de ser implementada. O software foi desenvolvido em plataforma Java o que possibilita migração do mesmo para outros computadores com sistemas operacionais distintos sem a necessidade de uma recompilação, visto a alta portabilidade da plataforma.

Abaixo pode ser observada a interface do software de controle. Maiores detalhes da interface serão apresentados nas seções subsequentes.

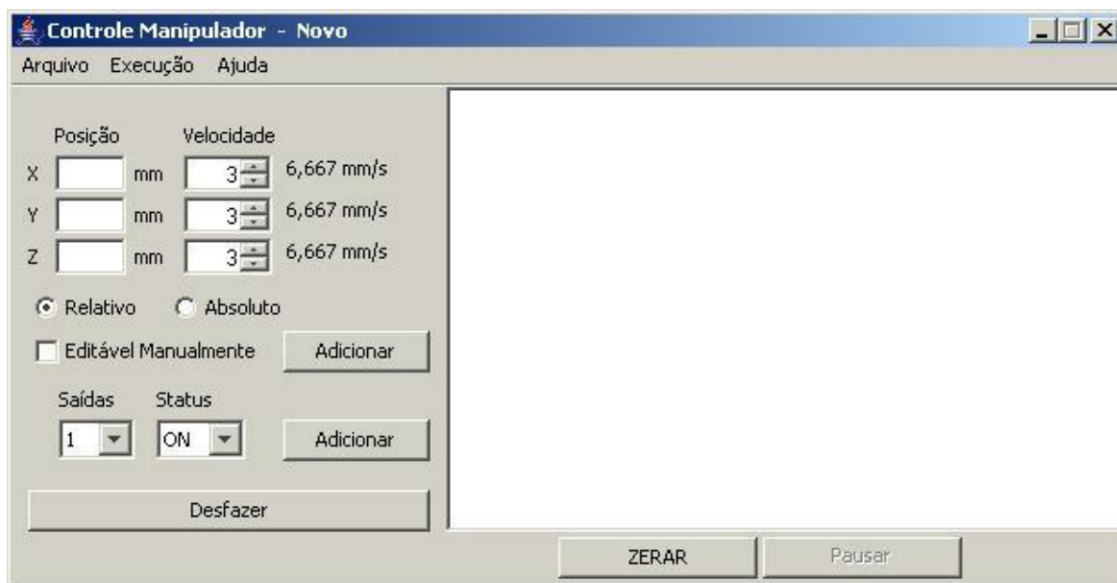


Figura 1

2. ESPECIFICAÇÕES DE HARDWARE

- Cartão de controle 46000 controlado via link RS232. Inclui:

- Botão reset que reinicializa o cartão caso seu programa esteja bloqueado;
- Alavanca de parada de emergência – interrompe os movimentos dos três eixos;
- Oito saídas a relé, podendo acionar cargas tanto em CC quanto em AC;
- Oito entradas – 5 a 50 V ativadas em nível baixo;
- Diodos emissores de luz que permitem monitoramento permanente das entradas, saídas, controles dos motores de passo e módulo de força de cinco volts.

- Módulo de força

- Manipulador

- Dimensões externas: 700 x 500 x 400 mm³;
- Três motores de passo:
 - Alimentação de 12 V;
 - Passo 1.8°;
 - 200 passos por revolução;
- Três sensores: fins de curso
- Deslocamentos: 0,02 mm/passo nos três eixos
- Velocidade máxima de deslocamento: 1000 passos/s (20 mm/s)
- Espaço de trabalho: 297 x 210 x 100 mm

3. SOFTWARE DE CONTROLE

3.1. Instalação da Máquina Virtual Java

Para a instalação do software, primeiramente deve ser instalada a Máquina Virtual Java de versão 1.5.0 ou superior juntamente com a COMMAPI, uma API de extensão da linguagem que permite interação com portas seriais e paralelas. Tanto a máquina virtual quanto sua API de extensão podem ser obtidas em <http://java.sun.com/>.

3.2. Descrição da Interface

Corpo da Interface: Tomemos a figura abaixo como base para descrever cada item da interface:

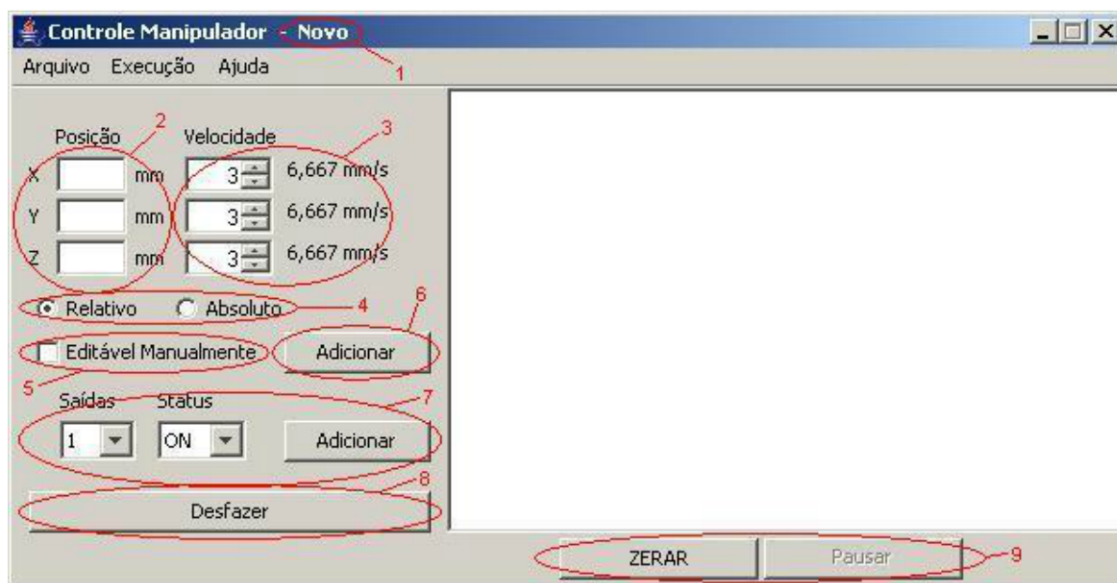


Figura 2

- 1 – Nome do arquivo aberto;
- 2 – Nestes campos devem ser digitados os deslocamentos com que se deseja programar o manipulador para movimentos nos diversos eixos;
- 3 – Controle de velocidade de deslocamento nos três eixos. O valor digitado é convertido em mm/s mais à direita (restrição do manipulador). Este resultado é obtido convertendo-se o valor do campo velocidade através da seguinte fórmula:

$$\text{velocidade} = \frac{\text{deslocamento em mm/passo} \times 1000}{\text{velocidade de máquina}} \quad (1)$$

sendo:

- velocidade – resultado em mm/s
- deslocamento em mm/passo = 0,02 mm (ver especificações de hardware)
- velocidade de máquina obtida no campo velocidade da interface

substituindo-se estes valores na equação (1), temos:

$$\text{velocidade} = \frac{0,02 \times 1000}{\text{velocidade de máquina}} = \frac{20}{\text{velocidade de máquina}} \text{ mm/s} \quad (2)$$

- 4 – Determina qual será a espécie de deslocamento utilizada na programação.
 - Relativo – o manipulador desloca exatamente o que foi digitado nos campos posição (item 2) na velocidade escolhida nos campos velocidade (item 3);
 - Absoluto – o manipulador desloca para as coordenadas digitadas nos campos posição;
 Ex.: Supondo o manipulador na coordenada X = 100 mm e deslocamento em X de 20 mm. Em modo relativo o manipulador se moveria 20 mm indo para X = 120 mm enquanto em modo absoluto o mesmo retornaria para a posição 20 mm deslocando, assim, -80 mm;
- 5 – Permite fazer alterações manuais no campo de código fonte mais a direita da interface. Este campo não é editável por padrão a fim de estimular os programadores deixarem o próprio software gerar o código-fonte evitando erros de digitação;
- 6 – Adiciona ao código fonte um deslocamento indicado pelos campos “Posição” na velocidade indicada pelos campos “Velocidade”;
- 7 – Permite controlar as 08 saídas do cartão de controle ligando ou desligando-as. O botão adicionar gera uma instrução com este fim;
- 8 – Apaga a última linha adicionada ao código fonte;
- 9 – Botões de ação:
 - ZERAR/Executar – inicialmente é necessário ZERAR as coordenadas do manipulador para começar a execução de um programa qualquer. Após o zeramento, o texto do botão se torna executar e a manipulação pode ser iniciada;
 - Pausar/Passo Seguinte – há 2 formas de execução de um programa:
 - Direta – executa o programa inteiro;
 - Passo a passo – executa o programa linha a linha;
 -

Na execução direta é possível pausar o programa através deste controle. A instrução corrente é concluída e o manipulador aguarda até a pausa ser desfeita.

Na execução passo a passo o texto do botão muda para “Próximo passo” e se faz necessário o acionamento deste controle ao fim de cada instrução para dar continuidade ao programa.

Menus da Interface: Tomemos a figura abaixo como base para descrever os menus do aplicativo e as opções que cada um deles oferece:

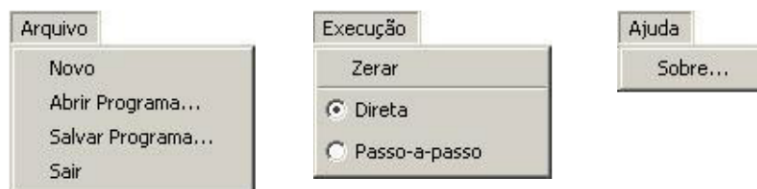


Figura 3

- Arquivo – possui opções gerais para controle de arquivos. Neste menu é possível criar novos programas, abrir existentes, salvar o atual e sair do aplicativo;

- Execução – apresenta opções de execução do programa. O manipulador pode ter suas coordenadas zeradas através deste menu e a execução pode ser configurada entre direta e passo a passo:
 - Direta – executa o programa inteiro;
 - Passo a passo – executa o programa linha a linha;
- Ajuda – apresenta uma opção para visualizar os dados do software e do autor: nome do software, versão, autor, endereço eletrônico;

3.3. Execução

Demonstraremos a execução de um programa simples supondo que os 3 eixos encontram-se zerados e que haja uma broca fixo ao manipulador acionada pela saída 1. Para isso, observemos o código-fonte abaixo transcrito:

```
1 - REL: XP 80   V 3      YP 50   V 3      ZP 30   V 3
2 - SAIDA 1 ON
3 - REL: ZP -30   V 15
4 - REL: ZP 30   V 3
5 - SAIDA 1 OFF
```

Abaixo a descrição do que cada linha do código realiza:

- 1 – Posiciona o manipulador no local onde será feito o furo deslocando os 3 eixos simultaneamente;
- 2 – A saída 1 é acionada ligando a broca;
- 3 – A broca desce 30 mm em velocidade lenta efetuando o furo;
- 4 – A broca retorna os 30 mm;
- 5 – Saída 1 desativada desligando a broca e concluindo o furo.

4. CONCLUSÃO

O software apresentou resultados satisfatórios no controle do manipulador RD46, substituindo por completo (falta à comunicação de todas as mensagens enviadas e/ou recebidas) o software original da Didacta Itália. A modernização na forma de controlar o manipulador se torna evidente ao se observar a nova interface com botões e caixas de texto atingindo o objetivo de tornar mais fácil e intuitiva a programação do mesmo para que mesmo as pessoas mais leigas e sem um contato mais aprofundado com computação, possam utilizar o manipulador em aulas laboratoriais.

Continuação deste trabalho poderá ocorrer com a utilização do ambiente de programação em conjunto com o módulo de força e assim poder acionar/ controlar outros dispositivos baseados em motores de passo (no máximo três).

Os testes de campo serão realizados pelos alunos da disciplina de robótica do curso de Tecnologia em Automação Industrial (programação de robôs) e assim a robustez do sistema e os possíveis “bugs” existentes poderão ser verificados em condições reais de utilização.