

## **Introdução à programação em TP na consola do manipulador Fanuc**

A consola de programação permite a construção da maioria dos programas necessários aquando da utilização do manipulador em ambiente industrial.

### **Aviso de Segurança**

Trabalhar com este robô pode ser bastante perigoso se não se tiver o devido cuidado. Nunca se deve utilizar uma velocidade elevada quando se programa. Os testes devem ser feitos em velocidade muito baixas.

### **Linguagem TP**

A linguagem TP é bastante simples sendo a sua estrutura básica a seguinte:

#### **J P[1] 100% FINE**

Em que J significa que o manipulador se vai mover para o ponto P[1] utilizando a trajectória mais fácil (menor movimento de juntas). O valor da percentagem é o da velocidade que o manipulador terá ao ir para P[1]. Estas instruções indicam uma movimentação entre dois pontos, aquele em que o robô está e o ponto definido em P[1] neste caso.

Para além do movimento de juntas existem ainda movimentos lineares ou circulares entre pontos.

O argumento FINE indica que o ponto será alcançado com precisão. No caso de ser, por exemplo, um ponto de passagem pode ser utilizado o CNT que permite que o robô não pare nesse ponto e passe “perto” dele.

Para definir pontos de passagem é necessário fazer o chamado TOUCHUP. Este implica que o utilizador coloque o robô na posição P desejada e depois grave essa posição.

<b>Exemplo 1 – Definir dois pontos de trabalho e executar Trajectórias J e L entre eles.</b>
--

Utilizando o manual *jog* armazenar dois pontos P[1] e P[2] com o comando TOUCHUP. No mesmo programa fazer um ciclo (utilizar os comandos LABEL e GOTO) em que o robô vai de P[1] para P[2] com um movimento de juntas e de P[2] para P[1] em trajectória linear.

### Exemplo 2 – Adquirir uma coordenada de um ponto para um registo

Neste exemplo pretende-se adaptar o programa anterior de modo a criar um novo ponto P[3], que tenha os valores de (X, Y) de P[1] e o valor de Z de P[2].

Para isso é necessário guardar num ponto de referência os valores (X,Y,Z) dos pontos:

$$PR[70] = LPOS$$

Depois de executado este comando, PR[70] passa a ser uma estrutura com 6 campos correspondentes às coordenadas (X,Y,Z) e às orientações do *end effector*.

Para guardar um destes valores numa variável faz-se:

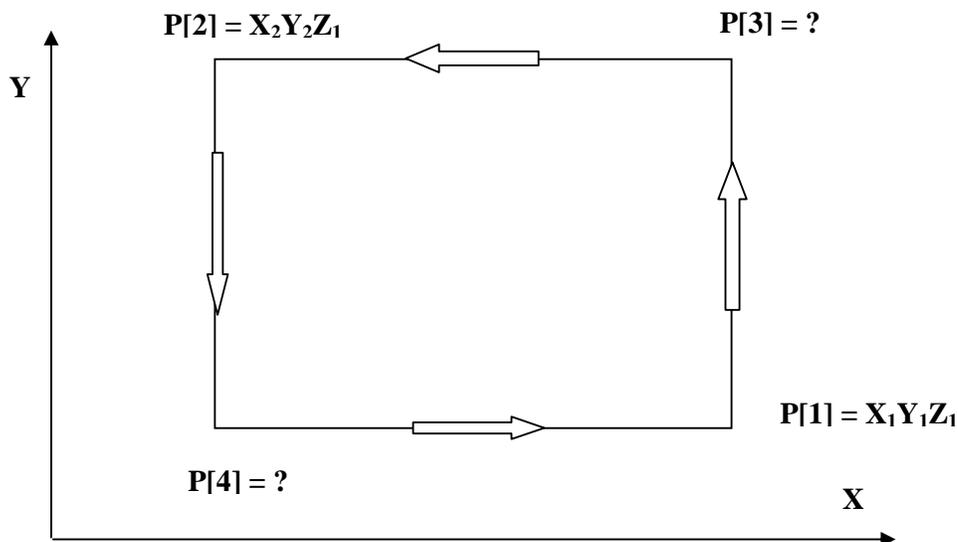
$$R[70] = PR[70, 3]$$

Depois de criar o P[3], acrescentar ao programa uma linha para ir até P[3]

### Exemplo 3 – Fazer a trajectória de um rectângulo perto da mesa de trabalho fazendo TOUCHUP a dois pontos apenas

Utilizando as funcionalidades atrás descritas, pretende-se fazer um programa que ponha o robô a executar um rectângulo no plano XY na mesa de trabalho.

Considerando a seguinte figura:



Pretende-se, fazendo o TOUCHUP a apenas P[1] e P[2], executar a trajectória representada.