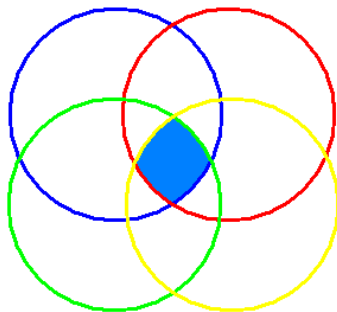




Introdução à Linguagem de Programação PDL2

(Parte 3)

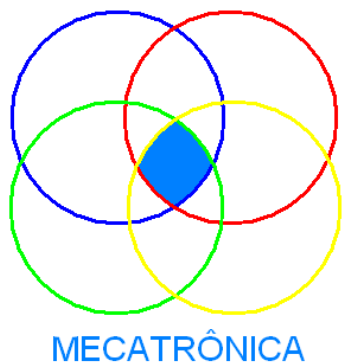


MECATRÔNICA





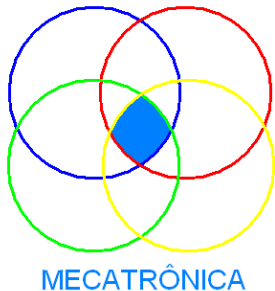
Comando de Movimentação na Linguagem PDL2



Linguagem PDL2 – Instruções

➤ MOVE

Inicia um comando de movimentação.

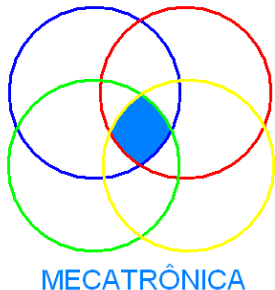


Funções de movimentação

```
MOVE <arm_clause> <traj_clause> dest_clause <opt_clauses>  
      <sync_clause>
```

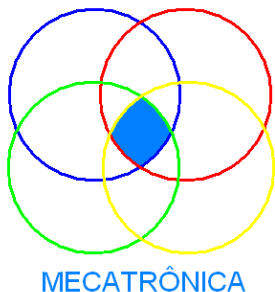
Se for necessário continuar na linha seguinte, é preciso usar “virgula” ao final da linha, após um argumento. Então, é preciso usar ENDMOVE no fim do comando.

```
PROGRAM armtest PROG_ARM=1  
.  
.  
.  
BEGIN  
  MOVE TO perch                -- moves arm 1  
  MOVE ARM[2] TO normal        -- moves arm 2  
END armtest
```



Campo ARM

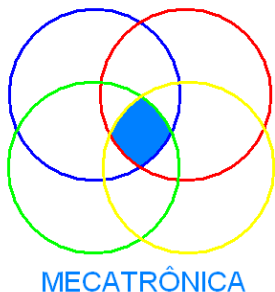
- Múltiplos braços podem ser controlados por um único programa PDL2.
- Se somente um braço será acionado o campo ARM não precisa ser especificado.
- Se o campo ARM não for incluído, o braço padrão será usado.
- O programador pode designar o braço padrão, como no exemplo anterior.



Campo TRAJECTORY

```
MOVE <arm_clause> <traj_clause> dest_clause <opt_clauses>  
      <sync_clause>
```

- A trajetória pode ser especificada ou associada às constantes predefinidas JOINT, LINEAR, CIRCULAR, ou associada com a variável predefinida \$MOVE_TYPE.



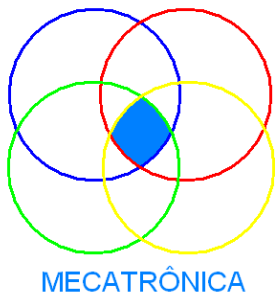
Campo TRAJECTORY

- A trajetória pode ser especificada ou associada às constantes predefinidas JOINT, LINEAR, CIRCULAR, ou associada com a variável predefinida \$MOVE_TYPE.

Ex.: MOVE LINEAR TO p1

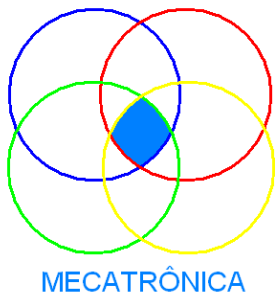
- O padrão para \$MOVE_TYPE é JOINT

```
$MOVE_TYPE := JOINT      -- assigns modal value
MOVE TO perch            -- joint move
MOVE LINEAR TO slot      -- linear move
MOVE TO perch            -- joint move
```



Campo TRAJECTORY

- Um movimento do tipo JOINT irá coordenar as juntas do braço de forma que iniciem e terminem seus movimentos ao mesmo tempo.
- Um movimento com trajetória LINEAR irá mover o ponto virtual da ferramenta do braço robótico por uma linha reta do ponto inicial até o ponto final.
- Uma trajetória CIRCULAR irá mover o ponto virtual da ferramenta segundo um arco.

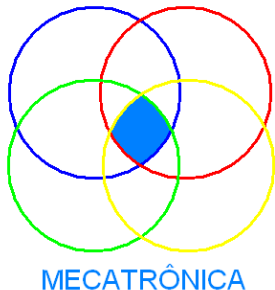


Campo DESTINATION

```
MOVE <arm_clause> <traj_clause> dest_clause <opt_clauses>  
    <sync_clause>
```

De acordo com o tipo de movimento, especifica o destino.

- **MOVE TO**
TO || destination | joint_list || <VIA_clause>
- **MOVE NEAR**
NEAR destination BY distance
- **MOVE AWAY**
AWAY distance
- **MOVE RELATIVE**
RELATIVE vector IN frame
- **MOVE ABOUT**
ABOUT vector BY distance IN frame
- **MOVE BY**
BY relative_joint_list
- **MOVE FOR**
FOR distance TO destination



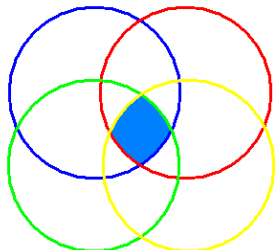
MOVE TO

➤ O destino pode ser alguma expressão resultando em um dos tipos:

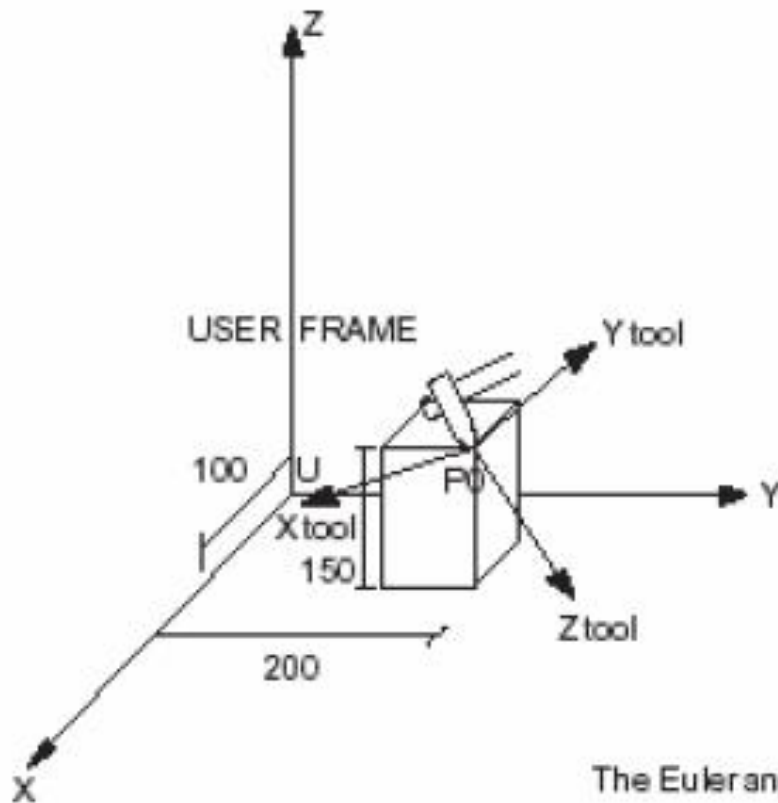
- POSITION
- JOINTPOS
- XTNDPOS

Por exemplo:

```
MOVE LINEAR TO POS (x, y, z, e1, e2, e3, config)  
MOVE TO perch  
MOVE TO home
```



Ângulos de Euler de rotação



```
PROGRAM posttest
```

```
VAR P0
```

```
BEGIN
```

```
P0:=POS(100,200,150,90,150,0,'')
```

location
components

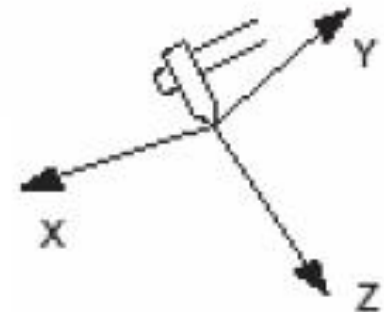
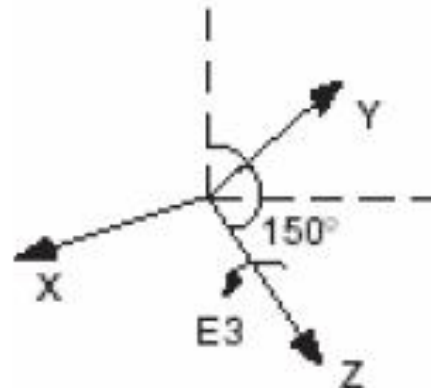
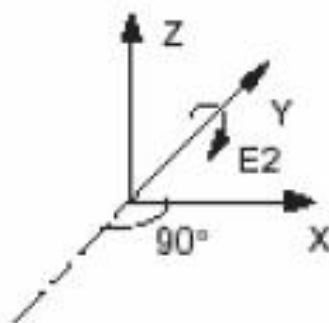
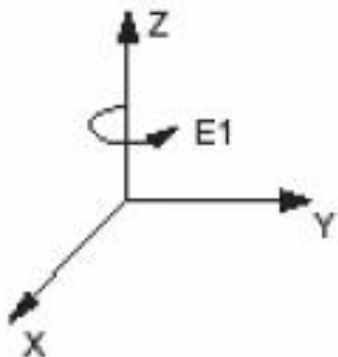
orientation
components

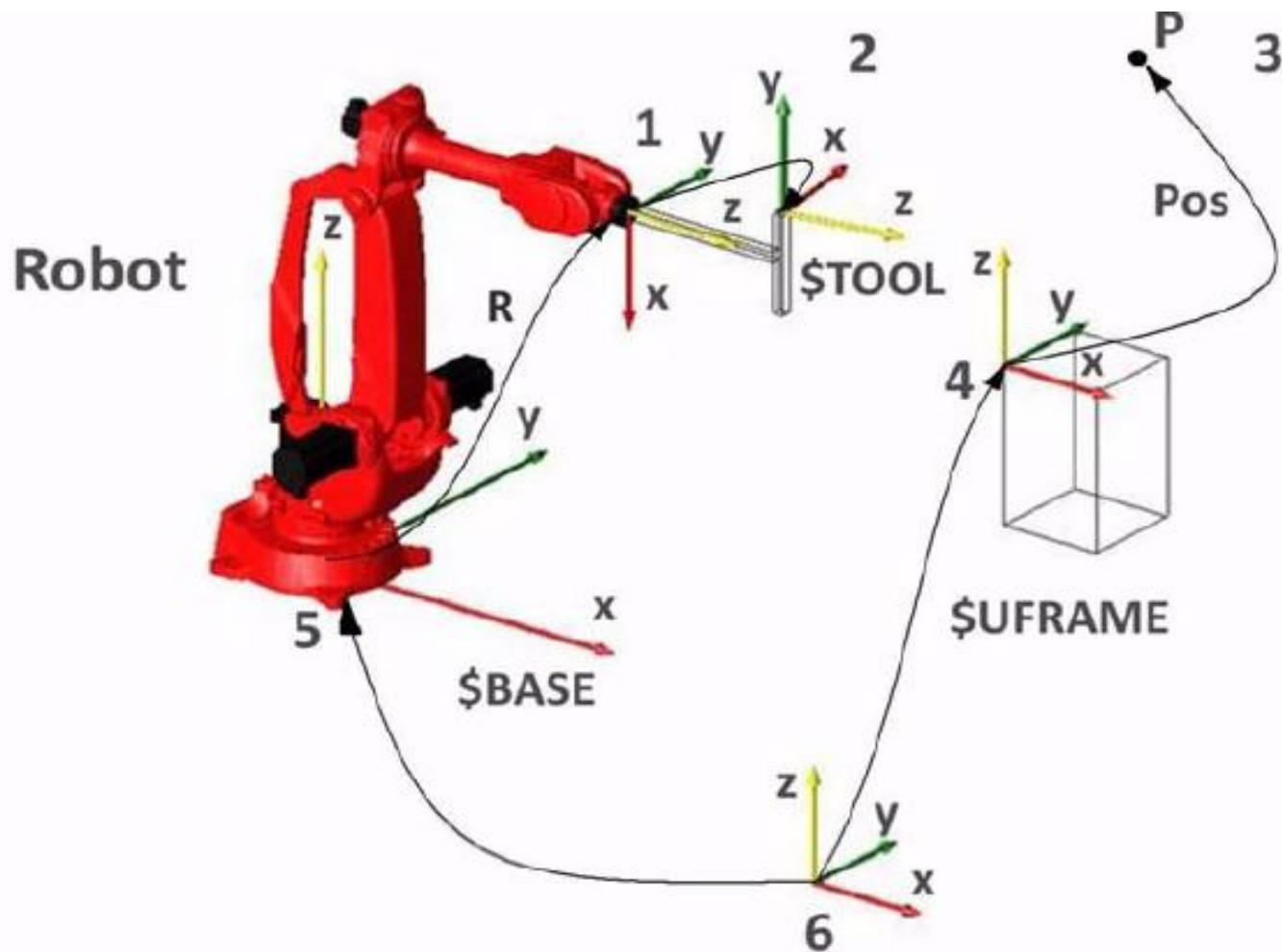
config.
components

```
END posttest
```

The location components are the three distances of P0 with respect to the user frame origin (U).

The Euler angles are defined as follow.





- | | | |
|------------------|----------------|---------------------|
| 1 - Flange Frame | 2 - Tool Frame | 3 - Taught Position |
| 4 - User Frame | 5 - Base Frame | 6 - World Frame |

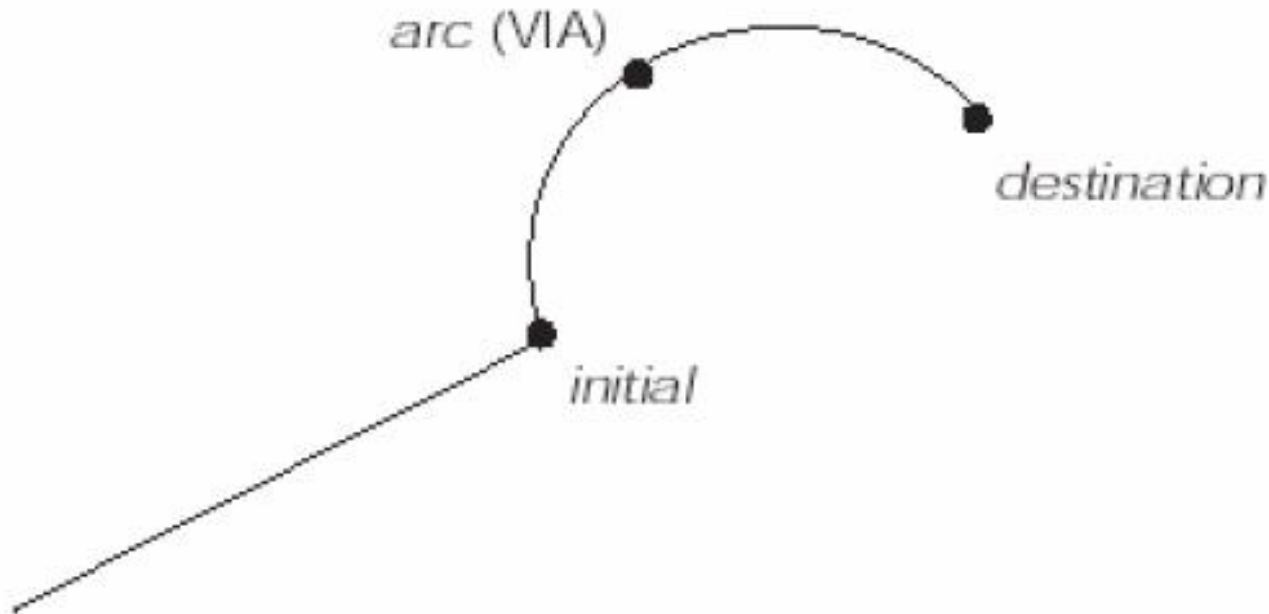
The world frame is predefined for each arm. The programmer can define the base frame (**\$BASE**) as a position, relative to the world frame. The programmer also can define the end-of-arm tooling (**\$TOOL**) as a position, relative to the faceplate of the arm. **\$UFRAME** is a transformation used to describe the position of the workpiece with respect to the world.

VIA Clause

- É usado para especificar uma posição pela qual o braço passe entre a posição inicial e o destino.
- É normalmente usado para definir um arco de um movimento circular.

```
MOVE TO initial
```

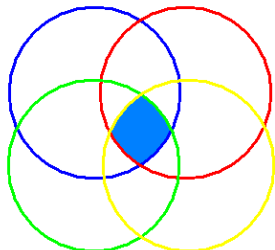
```
MOVE CIRCULAR TO destination VIA arc
```



MOVE NEAR

- Permite ao operador posicionar a ferramenta a uma distância (expressa em Real) de uma posição de destino.
- A distância é medida em mm ao longo do sentido negativo do vetor de aproximação da ferramenta.

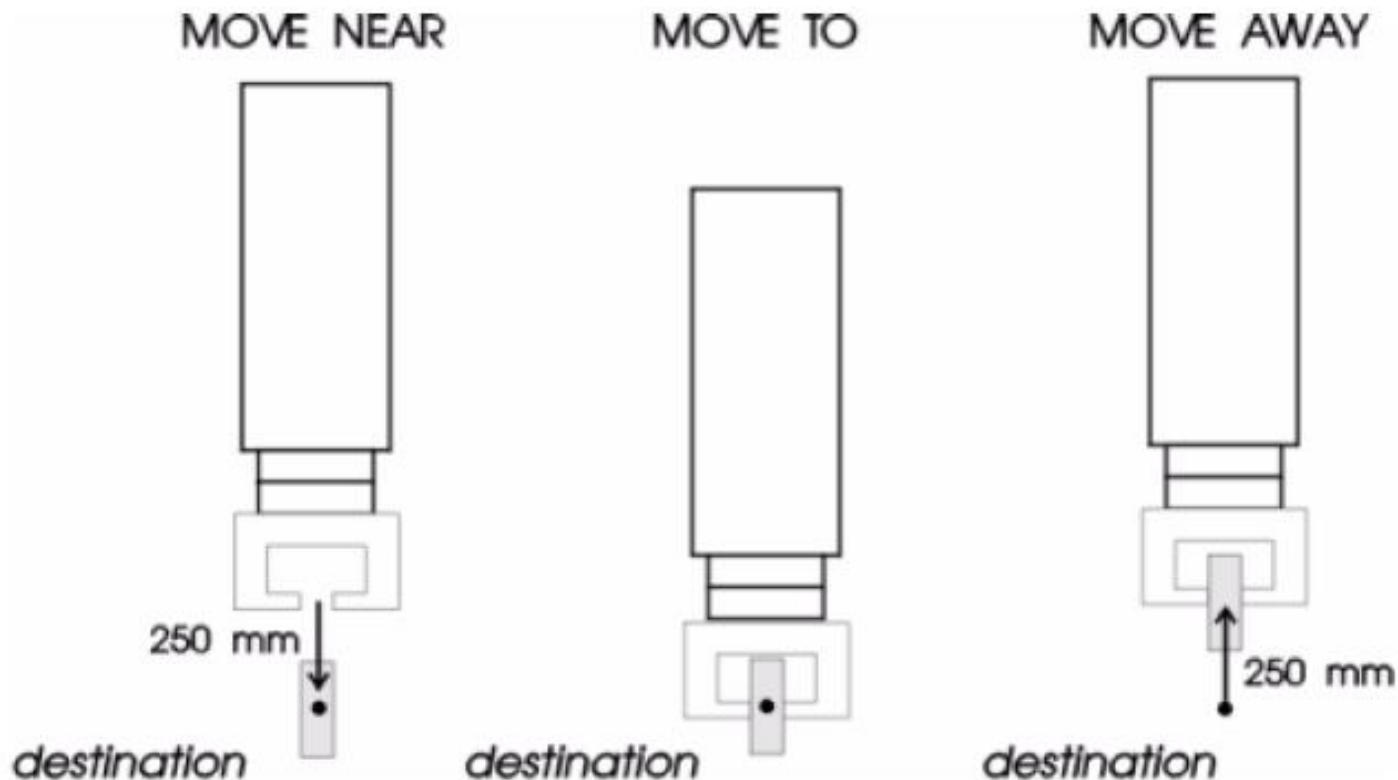
MOVE NEAR *destination* BY 250.0



MOVE AWAY

➤ Permite ao operador mover a ferramenta para uma distância em mm em relação ao sentido negativo do vetor de aproximação da ferramenta.

MOVE AWAY 250.0

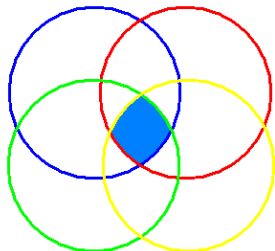
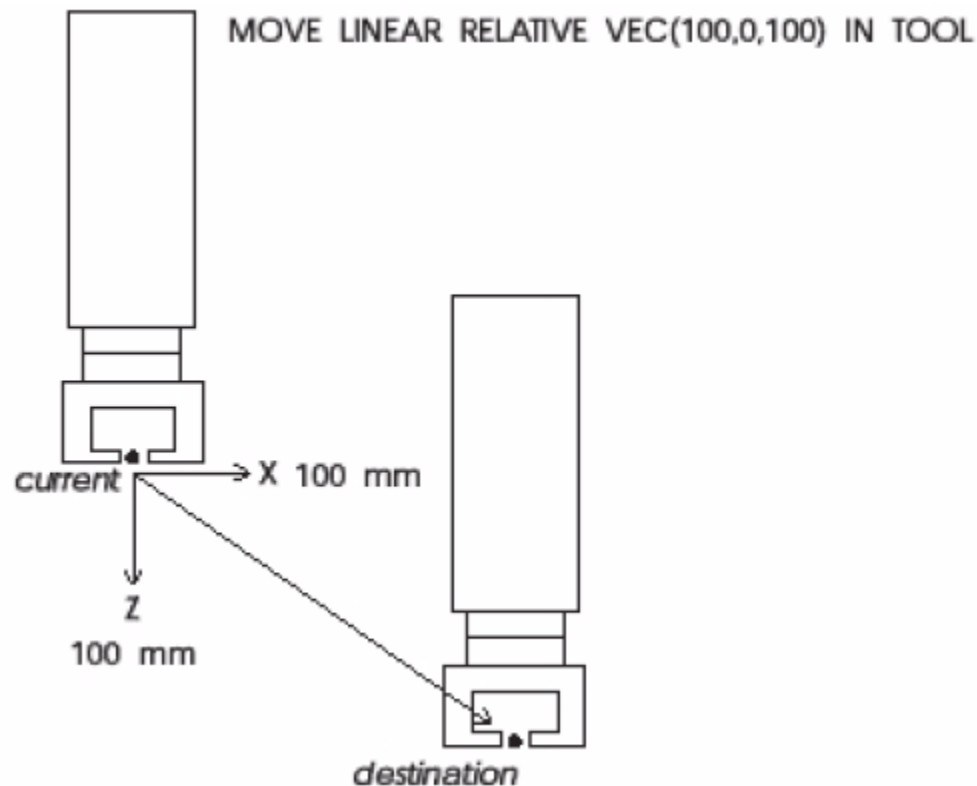


MOVE RELATIVE

- Permite ao operador indicar um destino, dado por um vetor, em relação à posição atual.

```
MOVE RELATIVE VEC(100, 0, 100) IN TOOL
```

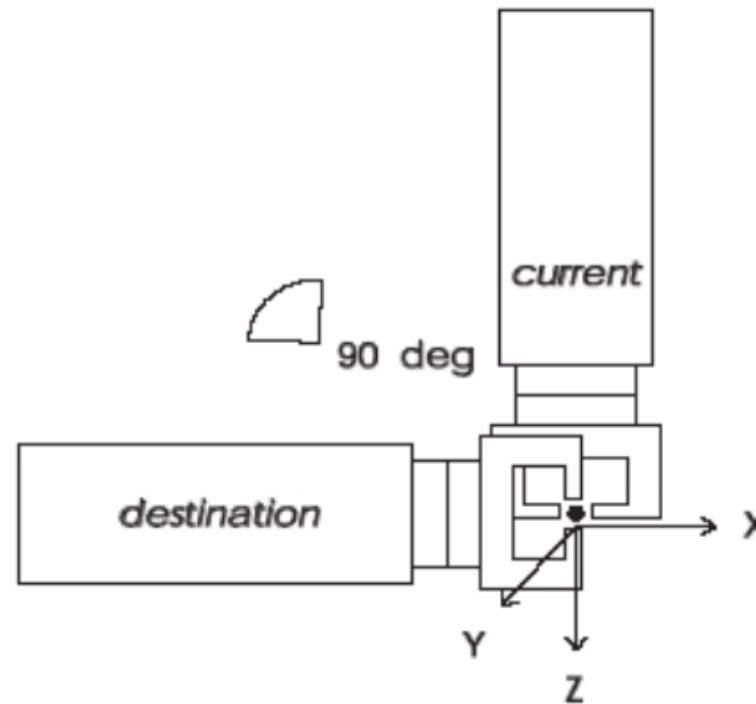
```
MOVE LINEAR RELATIVE VEC(100, 0, 100) IN TOOL
```



MOVE ABOUT

- Permite especificar um destino alcançado através da rotação da ferramenta de um ângulo em relação à pose inicial.

MOVE ABOUT VEC(0, 100, 0) BY 90 IN TOOL

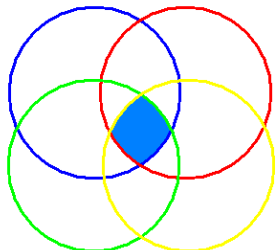


MOVE BY

- Permite especificar um destino como uma lista de expressões Reais, correspondendo a movimento (em graus ou em milímetros) incremental das juntas de um manipulador.

```
MOVE BY {alpha, beta, gamma, delta, omega}
```

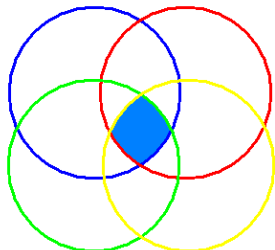
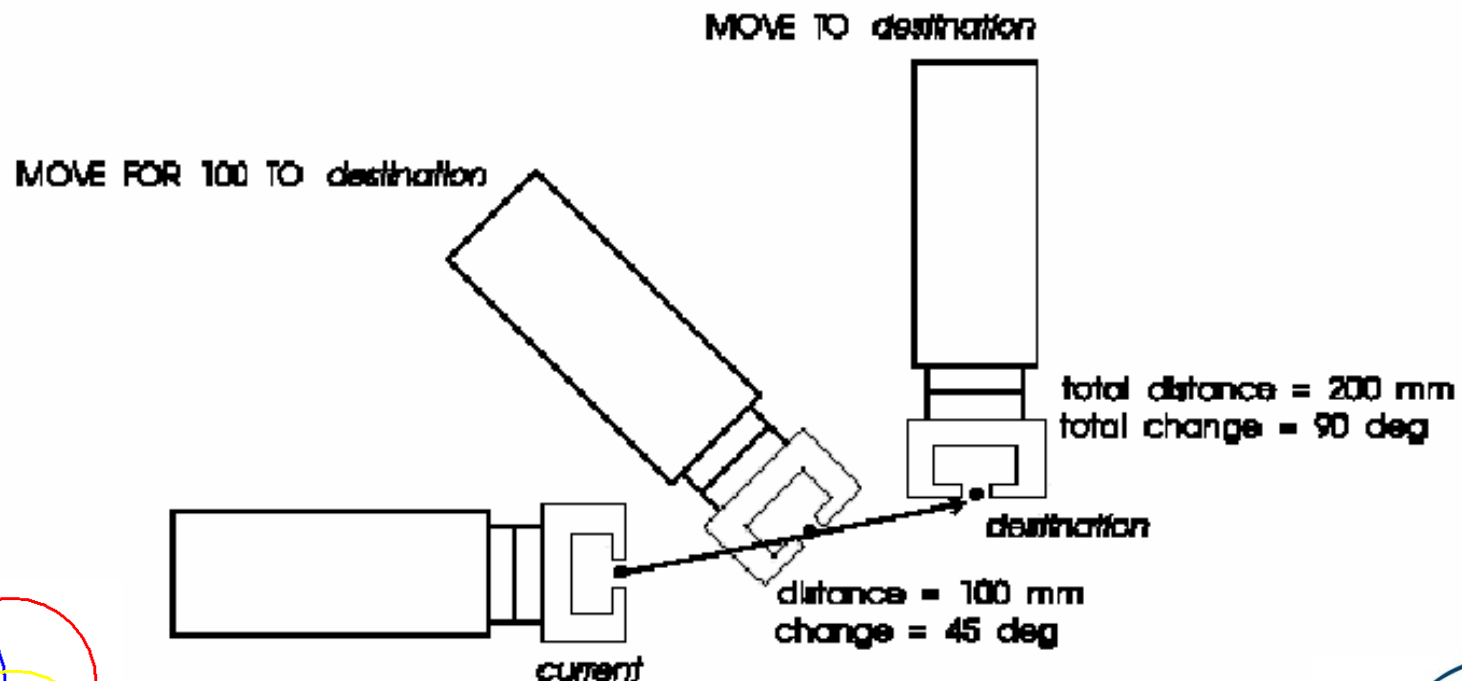
```
-- where alpha corresponds to joint 1, beta to joint 2, etc.
```



MOVE FOR

- Permite especificar um movimento parcial ao longo de uma trajetória especificada.

MOVE FOR *distance* TO *destination*



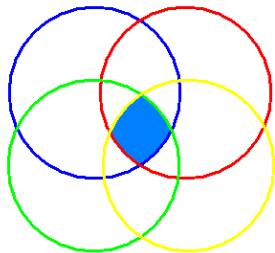
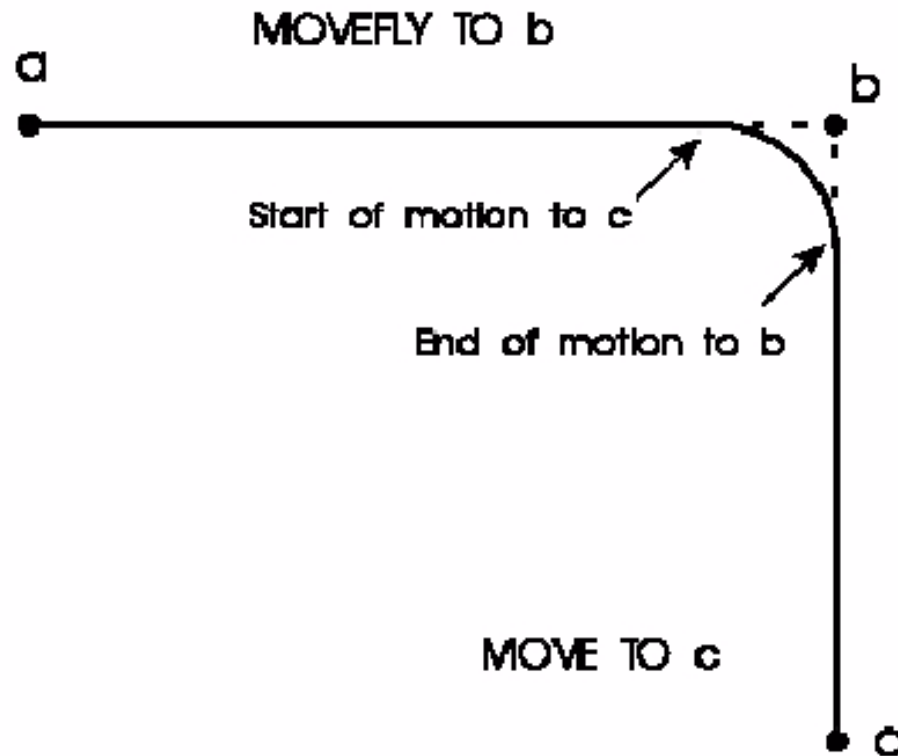
Movimento Contínuo (MOVEFLY)

- Permite mover o manipulador sem parada entre trechos de uma movimentação.

MOVE TO a

MOVEFLY TO b ADVANCE

MOVE TO c



MECATRÔNICA

Temporização e Sincronização

- Se o tempo requerido por um MOVEFLY é menor que o tempo gasto pelo interpretador para setar o próximo MOVE, o FLY não ocorrerá. Isso ocorre porque o sistema não obtém a informação que ele precisa a tempo de fazer o FLY.
- O FLY não terá efeito se houverem comandos adicionais entre o MOVEFLY e o próximo MOVE, que faça com que o interpretador demore em setar a próxima movimentação.