

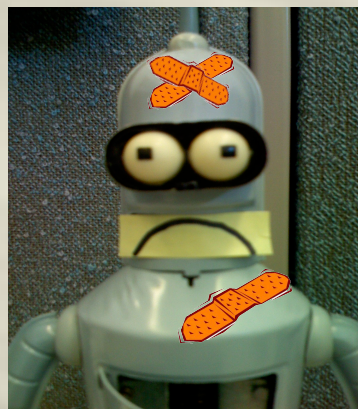
*Robotica – Robot Industriali e di Servizio*

*Lezione 7:  
La programmazione*

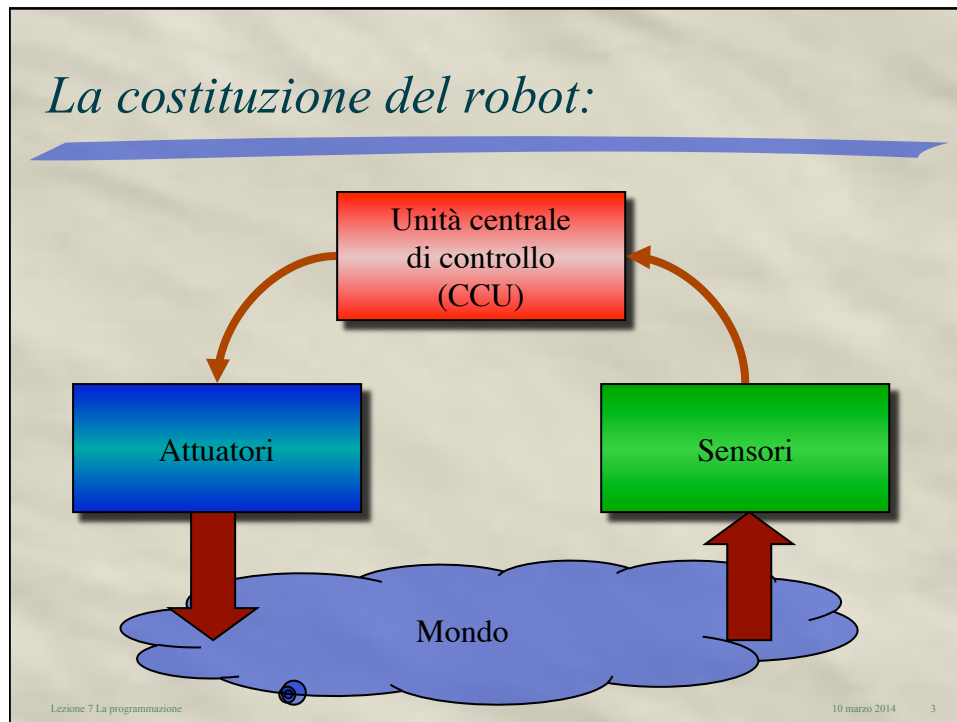
10 marzo 2014

*Intermezzo storico:*

- ⇒ I linguaggi di programmazione per robot sono nati negli anni '70
- ⇒ Quelli “belli” sono nati nelle Università
- ⇒ Stanford
- ⇒ Politecnico di Milano
- ⇒ Karlsruhe



Lezione 7 La programmazione

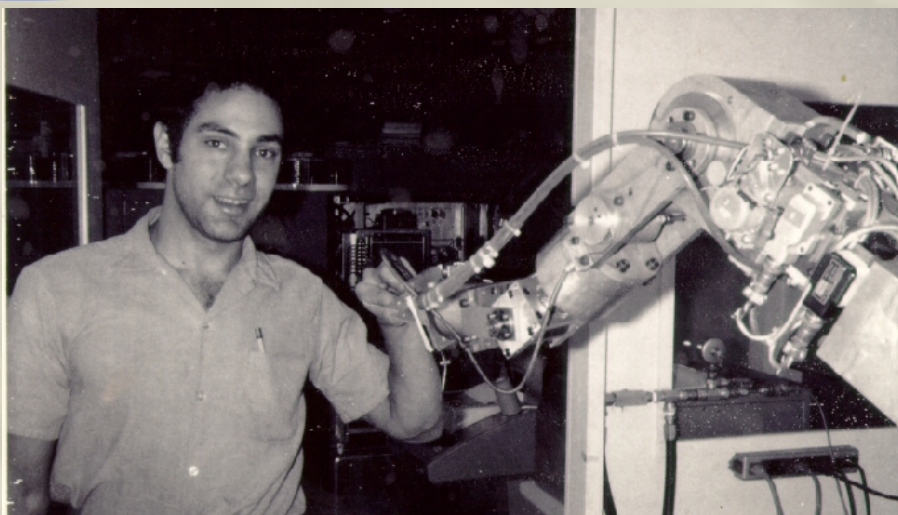


### Linguaggi di programmazione esplicita: concetti generali

- ⇒ Un manipolatore è del tutto simile ad un calcolatore
- ⇒ Ovviamente, deve avere un set di istruzioni particolare
  - Istruzioni di assegnamento, aritmetico-logiche, ecc.
  - Istruzioni di controllo
  - Istruzioni di I/O
  - Istruzioni “robot oriented”
    - Move
    - Test
    - Act
    - Sense
    - Esecuzione parallela (multithreading)
    - ...
- ⇒ Possiamo definire un calcolatore “virtuale” con queste caratteristiche

Lezione 7 La programmazione 10 marzo 2014 4

*Tutto cominciò tanti anni fa...*



Lezione 7 La programmazione

10 marzo 2014 5

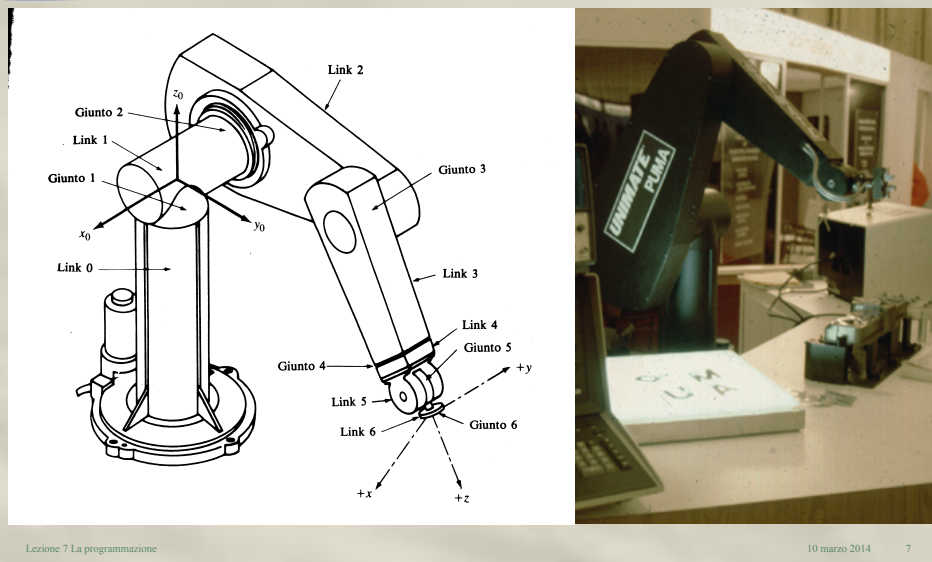
*Alla Stanford University...*



Lezione 7 La programmazione

10 marzo 2014 6

## *Il robot antropomorfo più famoso:*



Lezione 7 La programmazione

10 marzo 2014 7

## *I movimenti del manipolatore*

- ⇒ Controllo Point-To-Point (PTP):
  - Si specificano le coordinate del punto da raggiungere
- ⇒ Controllo Continuous Path (CP):
  - Si specifica il punto da raggiungere e la traiettoria da seguire per raggiungerlo
    - Semplice (linee rette)
    - Complessa (spline o altri metodi matematici)
    - Qualunque (robot di verniciatura)
- ⇒ Alcuni robot PTP consentono (per loro natura) alcuni movimenti in linea retta

Lezione 7 La programmazione

10 marzo 2014 8

## *Programmazione di un manipolatore industriale: classificazione*

- ⇒ Insegnargli dei movimenti che esso dovrà poi ripetere
- ⇒ Metodi fondamentali:
  - Espliciti
    - Teach in (teaching by showing, teaching by doing)
    - Linguaggi di programmazione esplicita
    - Misti
  - Impliciti
    - Sistemi CAD
    - Pianificatori automatici

Lezione 7 La programmazione

10 marzo 2014 9

## *C'è un'altra classificazione:*

- ⇒ Programmazione “on line”
  - Richiede la presenza fisica del manipolatore
- ⇒ Programmazione “off line”
  - Viene fatta a tavolino, o usando altri sistemi (simulatori, ecc.)

Lezione 7 La programmazione

10 marzo 2014 10

## Un'altra definizione: il telemanipolatore

⇒ Questo NON è programmabile!

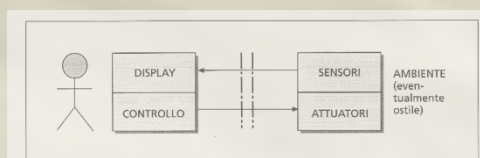


Lezione 7 La programmazione

10 marzo 2014 11

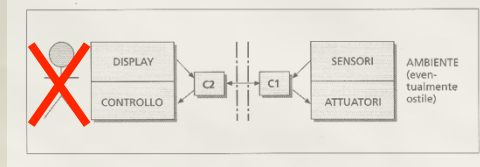
## Telemanipolatore vs. manipolatore:

Telemanipolatore:



Manipolatore:

L'uomo deve poter essere rimosso



Lezione 7 La programmazione

10 marzo 2014 12

## *I metodi "teach in"*

### ⇒ Obiettivo:

- ① far muovere manualmente il robot registrandone i movimenti
- ② "riprodurre" la registrazione

### ⇒ Metodi:

- Teach pendant
- Leading by nose (solo con azionamenti reversibili)
- Apprentice

### ⇒ Vantaggi:

- Possibilità di descrivere movimenti complicati
- Scarso addestramento del personale

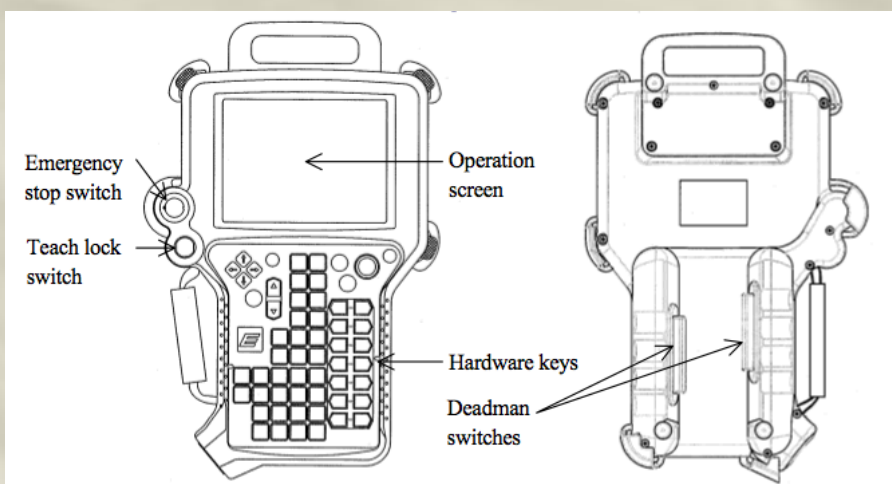
### ⇒ Svantaggi:

- Impossibilità di descrivere lavori complessi

Lezione 7 La programmazione

10 marzo 2014 13

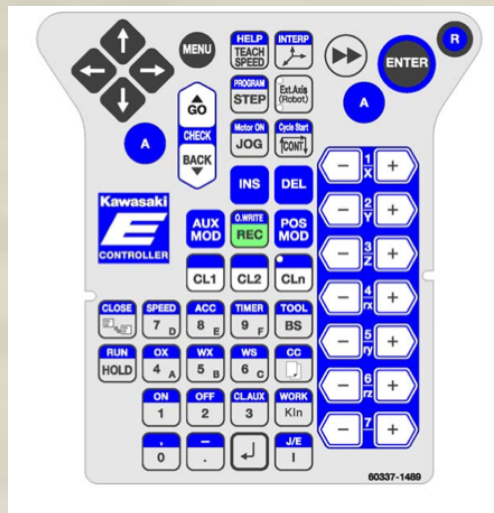
## *Ecco un "teach pendant"*



Lezione 7 La programmazione

10 marzo 2014 14

## La tastiera:

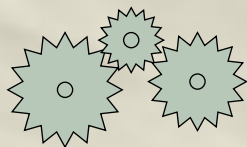


Lezione 7 La programmazione

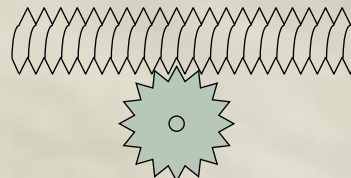
10 marzo 2014 15

## Leading by nose

- ⇒ I motori vengono spenti. Il robot viene spostato a mano, e si utilizzano i suoi encoder per rilevarne continuamente la posizione.
- ⇒ Possibile solo con azionamenti reversibili!



**Sì**

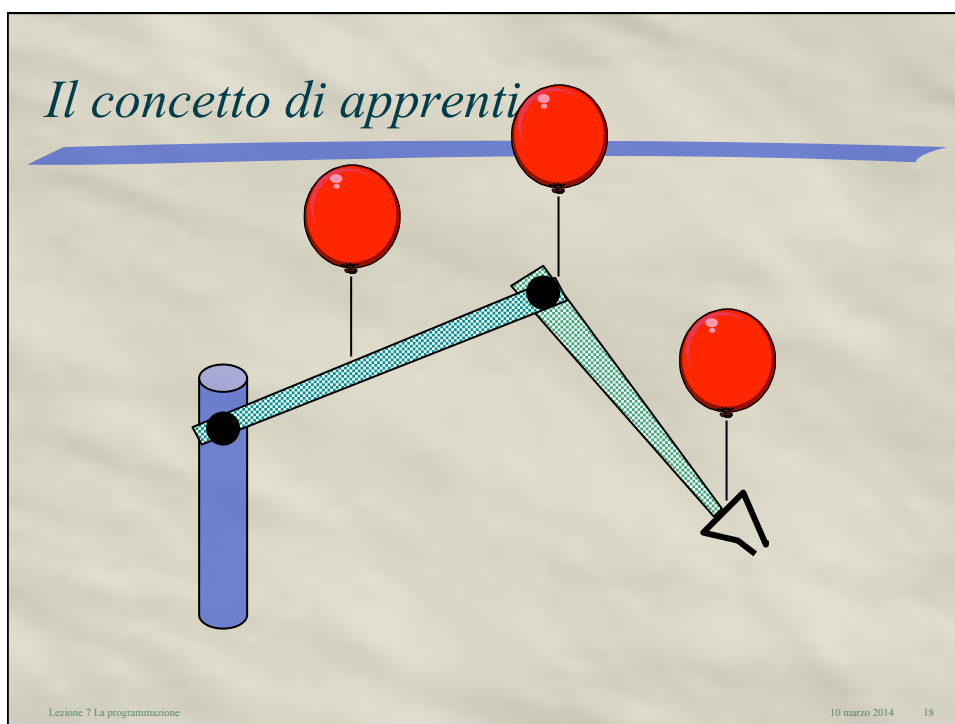
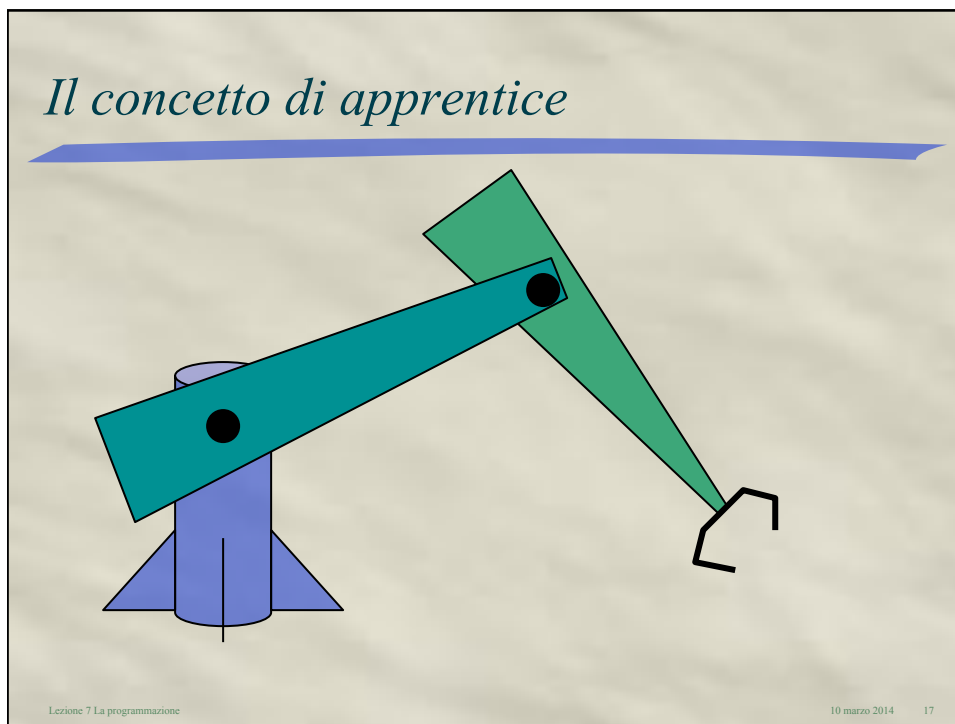


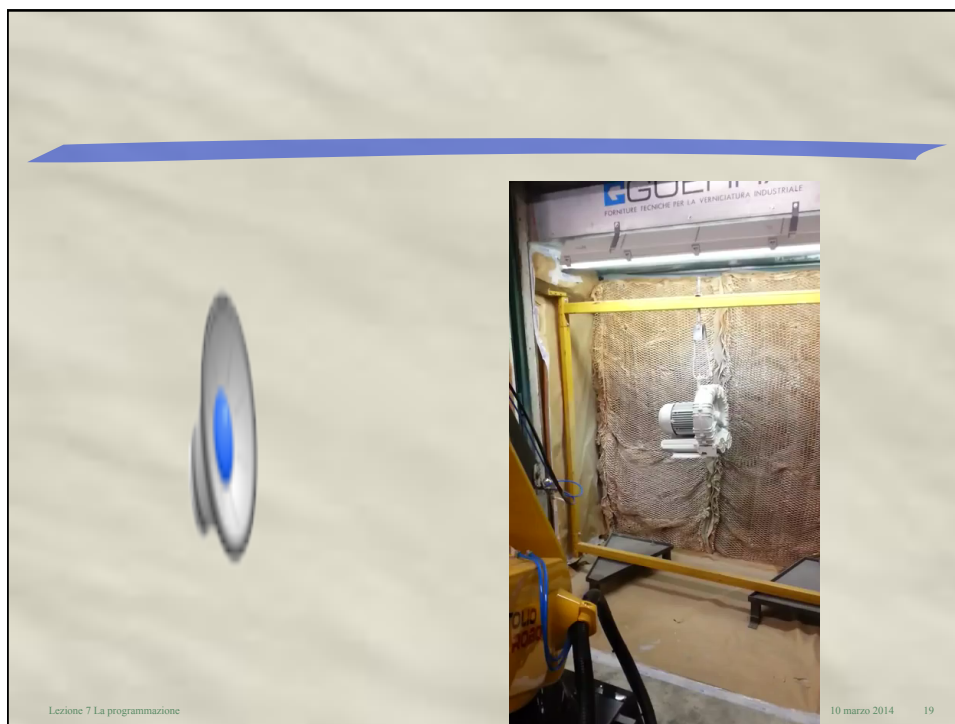
**No!**

Lezione 7 La programmazione

10 marzo 2014 16







### *Quattro esempi di programmi per robot:*

- ⇒ Linguaggi derivati dalle macchine a controllo numerico
- ⇒ Linguaggi derivati dai linguaggi di programmazione “classici” (Basic, Pascal, ecc.)

## *Un programma orribile!*

```

PROG 00001
N0000 G98 S01,1 S63,8,1 S63.9,3 S71,1 S00
N0001 F8 S00
N0002 F8 S97,1 S00
N0003 G01 F500 S00
N0004 G98 S60,20 S70,1 S00
N0005 G01 F500 S00
N0006 F8 S00
N0007 G01 F500 S00
N0008 G98 S60,20 S71,1 S00
N0009 F8 S60,60 S00
N0010 F8 S00
N0011 G01 F500 S00
N0012 G98 S60,20 S70,1 S00
N0013 G01 F500 S00
N0014 F8 S00
N0015 G01 F500 S00
N0016 G98 S60,20 S71,1 S00
N0017 G01 F500 S10,1 S00
N0018 G98 S17,1,50 S30,1 S32,2 S00
N0019 F8 S97,2 S99
    
```

Lezione 7 La programmazione

10 marzo 2014 21

## *Un programma più leggibile*

```

PROG 1
10 VELOC 500 mm/sec MAX VELOC 1000 mm/sec
20 TCP 2
30 RECT COORD
40 POS V = 100%
50 SET R5 = 0
60 POS V = 100%
70 POS 50% FINE 1
80 WAIT TIME 2s
90 GRIPPER CLOSE
100 POS V = 50%
110 POS V = 100%
120 POS V = 50% FINE
130 WAIT TIME 2s
140 GRIPPER OPEN
150 POS V = 100%
160 WAIT TIME 6s
170 POS V = 50% FINE
180 WAIT TIME 2s
190 GRIPPER CLOSE
200 POS 50%
210 POS 100%
220 POS 50% FINE
230 WAIT TIME 2s
240 GRIPPER OPEN
250 POS 50%
260 POS 100%
270 ADD 1 TO R5
280 JUMP TO 60 IF R5<50
290 END
    
```

Lezione 7 La programmazione

10 marzo 2014 22

## Un programma leggibile (1)

```

. HERE CASTING
X/JT1   Y/JT2   Z/JT3   O/JT4   A/JT5   T/JT6
54.35   236.78   -29.03   -202.6   78.90   113.76

. HERE BANDSAW
X/JT1   Y/JT2   Z/JT3   O/JT4   A/JT5   T/JT6
305.5   45.78   35.8    -56.8    -34.98   78.21

. HERE CONVEYOR
X/JT1   Y/JT2   Z/JT3   O/JT5   A/JT6   T/JT6
205.45  -389.4   25.97   -178.4   78.63   170.3

.   EDIT BANDSAW FEEDING .1
.   EXECUTE BANDSAW FEEDING .1,50
1.? SETI COUNT = 0
2.? 10 APPRO CASTING, 25
3.? MOVE 'S' CASTING
4.? WAIT 2
5.? CLOSE I
    
```

Lezione 7 La programmazione

10 marzo 2014 23

## Un programma leggibile (2)

```

6.? DEPART 'S', 50
7.? APPRO BANDSAW, 100
8.? MOVE 'S', BANDSAW
9.? WAIT 2
10.? OPEN I
11.? DEPART 'S', 350
12.? WAIT 6
13.? APPRO BANDSAW, 100
14.? MOVE 'S' BANDSAW
15.? WAIT 2
16.? CLOSE I
17.? DEPART 'S', 250
18.? APPRO CONVEYOR, 25
19.? MOVE 'S', CONVEYOR
20.? WAIT 2
21.? OPEN I
22.? DEPART 'S' 100
23.? SETI COUNT = COUNT +1
24.? TYPEI COUNT
25.? GOTO 10
26.?
    
```

Lezione 7 La programmazione

10 marzo 2014 24

## Un programma ottimo (1)

```

1 PROGRAM machine tool loading
2
3 VAR
4 PARK           : Position      : park position
5 CHUCK          : Position      : chuck position
6 CONVEYOR IN   : Position      : conveyor-in position
7 CONVEYOR OUT  : Position      : conveyor-out position
8 PART COUNT 1  : Integer        : counter variable
9
10 CONST
11 CHUCK OPEN = 1 : output signal to open chuck
12 CHUCK CLOSE = 2 : output signal to close chuck
13 STOP CYCLE = 3 : input signal to stop program
14
15 BEGIN
16 $SPEED = 500
17 $MOTYPE = linear
18 $TERM TYPE = coarse
19 PART COUNT 1 = 0
20 REPEAT
21     MOVE TO PARK
22     MOVE TO CHUCK
23     CLOSE HAND 1

```

Lezione 7 La programmazione

10 marzo 2014 25

## Un programma ottimo (2)

```

24     DOUT [1] = ON
25     DELAY 2000
26     MOVE TO PARK
27     MOVE TO CONVEYOR-OUT
28     OPEN HAND 1
29     DELAY 2000
30     MOVE TO CONVEYOR-IN
31     CLOSE HAND 1
32     DELAY 2000
33     MOVE TO PARK
34     MOVE TO CHUCK
35     DOUT [2] = ON
36     DELAY 2000
37     OPEN HAND 1
38     DELAY 2000
39     MOVE TO PARK
40     PART COUNT 1 = PART COUNT 1+1
41
42     UNTIL DIN [stop cycle] = ON
43
44     WRITE ('Total Components Machined', part count)
45
46     END machine tool loading

```

Lezione 7 La programmazione

10 marzo 2014 26

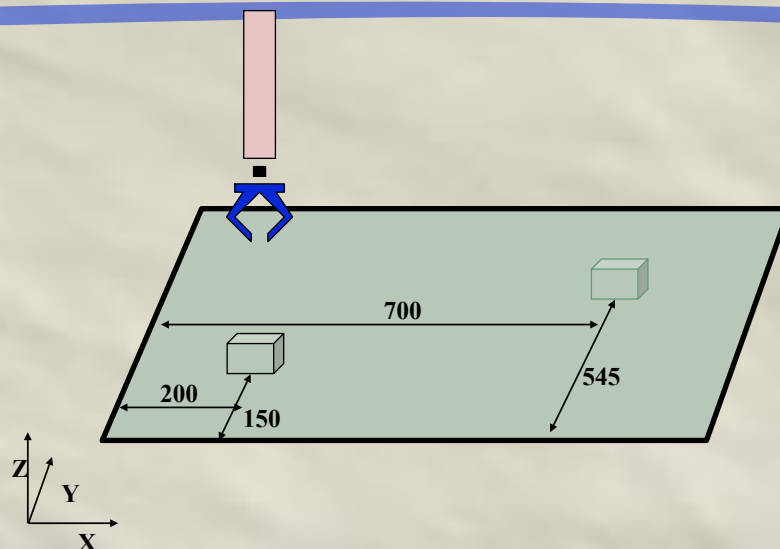
## *Le categorie di linguaggi*

- ⇒ Linguaggi per la programmazione esplicita
  - Linguaggi orientati ai giunti
  - Linguaggi orientati al manipolatore
  - Linguaggi orientati agli oggetti
- ⇒ Linguaggi per la programmazione implicita
  - Pianificatori automatici

Lezione 7 La programmazione

10 marzo 2014 27

## *Esempio di un programma*



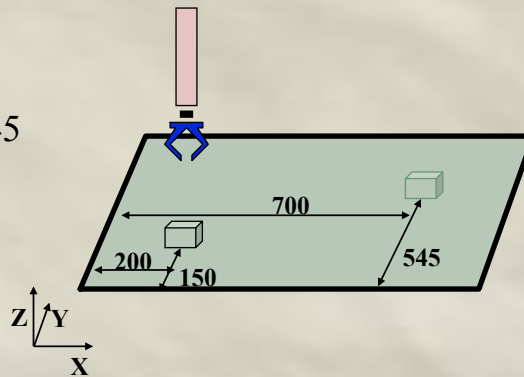
Lezione 7 La programmazione

10 marzo 2014 28

### *Il programma scritto in MAL*

```

10 MOVE X=200, Y= 150, Z=200
20 OPEN HAND
30 MOVE Z=20
40 CLOSE HAND
50 INCR Z=100
60 MOVE X=700,Y=545
70 INCR Z=-100
80 OPEN HAND
90 INCR Z=200
    
```



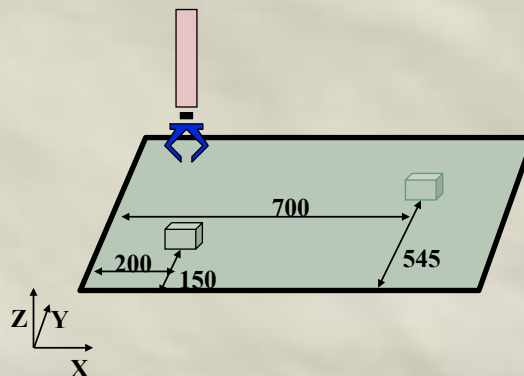
Lezione 7 La programmazione

10 marzo 2014 29

### *Il programma scritto in VAL*

```

APPRO P1, 100
OPENI
MOVES P1
CLOSEI
DEPARTS 100
APPROS P2, 100
MOVES P2
OPENI
DEPARTS 200
    
```



Lezione 7 La programmazione

10 marzo 2014 30

## Confronto fra i due sistemi

### ⇒ Linguaggi orientati ai giunti:

- Utilizzabili solo su alcuni robot (tipicamente cartesiani)
- Scarso controllo di traiettoria
- Semplici da implementare

### ⇒ Linguaggi orientati al robot:

- Indipendenti dal manipolatore
- Buon controllo di traiettoria (con possibilità di via points)
- Possibilità di spostare il sistema di riferimento (anche in tempo reale!)

## Un esempio in AS:

```
OPENI
HOME
JAPPRO #tutorial_centro, 100
LMOVE #tutorial_centro
CLOSEI
JAPPRO #tutorial_centro, 100
DRAW 68,68,0,0,0,0
DRAW 0,0,-100,0,0,0
OPENI
DRAW 0,0,100,0,0,0
HOME
```



## *Altri sistemi di programmazione*

### ⇒ Controllo in forza

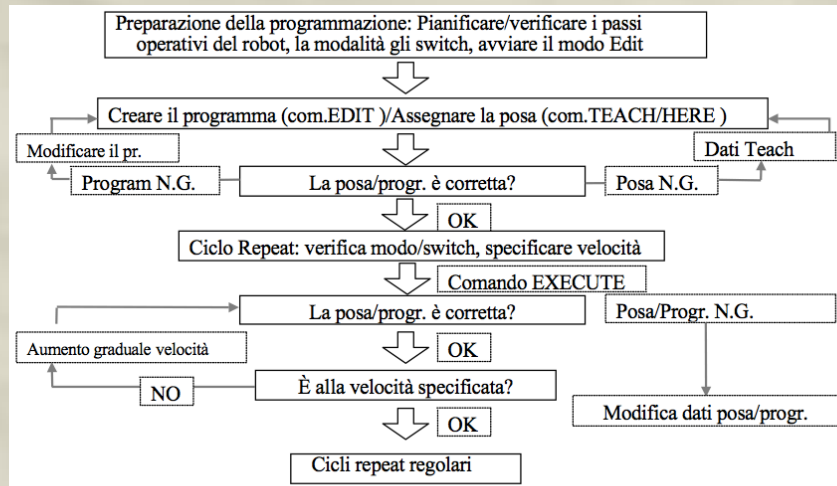
- Richiede appositi sensori
- Comporta notevoli problemi di controllo
- Non sempre la forza di reazione è normale alla superficie di contatto!
  - Cedevolezza selettivamente controllabile

## *In un linguaggio tipo VAL:*

### ⇒ Due fasi nella programmazione (iterate finché serve):

- Si “prendono i punti”
  - Si porta il braccio in ognuna delle posizioni significative per il lavoro da fare
- Si scrive il programma
  - Che fa muovere il braccio fra le varie posizioni secondo la sequenza voluta

### Più in dettaglio:



Lezione 7 La programmazione

10 marzo 2014 35

### La programmazione implicita

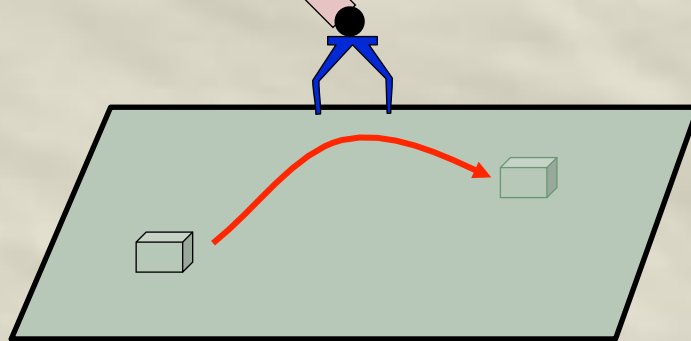
- ⇒ Non si specificano le azioni, ma il risultato
- ⇒ Occorre fornire al manipolatore tutte le informazioni:
  - Descrizione dei pezzi
  - Descrizione del risultato
  - Layout della cella
  - Descrizione del manipolatore(i)
  - Ne parleremo verso la fine del corso

Lezione 7 La programmazione

10 marzo 2014 36

*E adesso, vediamo di fare qualcosa di pratico*

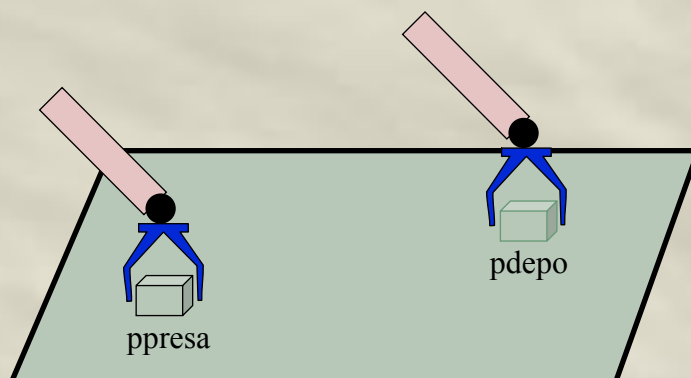
⇒ Il problema del pick and place



Lezione 7 La programmazione

10 marzo 2014 37

*Definiamo due pose:*



Lezione 7 La programmazione

10 marzo 2014 38

### *Il problema del pick and place*

⇒ Non possiamo muoverci così

The diagram shows a green rectangular table on a light grey floor. Three blue robots with black heads and pink arms are positioned around the table. The robot on the left is outside the table. The robot in the middle is on the table, labeled 'ppresa', and is holding a white cube. The robot on the right is on the table, labeled 'pdepo', and is holding a white cube. A blue horizontal bar is positioned above the text 'Non possiamo muoverci così'.

Lezione 7 La programmazione 10 marzo 2014 39

### *Il problema del pick and place*

⇒ E neppure così

The diagram is identical to the one above, showing three robots and a table. However, the robot labeled 'ppresa' is now holding a white cube, and the robot labeled 'pdepo' is also holding a white cube. A blue horizontal bar is positioned above the text 'E neppure così'.

Lezione 7 La programmazione 10 marzo 2014 40

### *Il problema del pick and place*

⇒ Dobbiamo muoverci così!

The diagram shows a green rectangular work area on a light grey background. Three robot arms, represented by blue grippers and pink shafts, are shown in different stages of a pick-and-place cycle. The first arm is on the left, approaching a cube labeled 'ppresa'. The second arm is in the center, holding a cube. The third arm is on the right, placing a cube labeled 'pdepo'. A blue horizontal bar is positioned above the text 'Dobbiamo muoverci così!'.

Lezione 7 La programmazione 10 marzo 2014 41

### *Fase 1: avvicinamento (approach)*

⇒ jappro ppresa, 100

The diagram shows a green rectangular work area on a light grey background. A robot arm, represented by a blue gripper and a pink shaft, is shown in the process of approaching a cube labeled 'ppresa'. A red coordinate system is overlaid on the scene, with the Z-axis pointing downwards towards the cube. Another cube is visible in the background on the right side of the work area.

Lezione 7 La programmazione 10 marzo 2014 42

### *Fase 2: Movimento finale (Imove)*

⇒ Imove ppresa

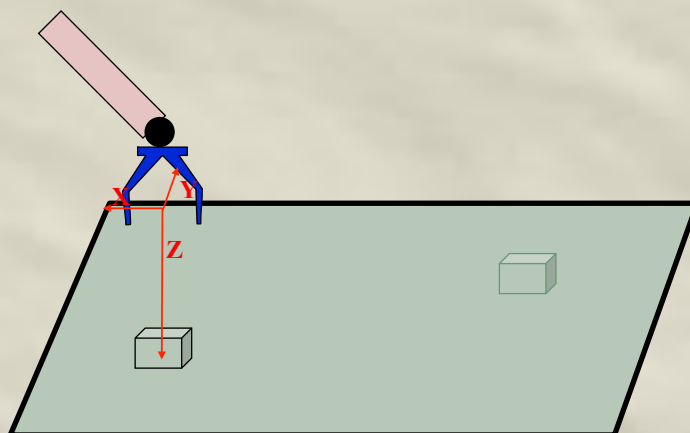


Lezione 7 La programmazione

10 marzo 2014 43

### *Fase 3: allontanamento (depart)*

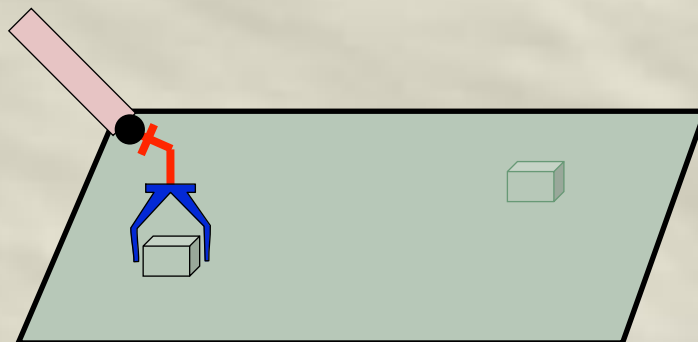
⇒ ldepart 100



Lezione 7 La programmazione

10 marzo 2014 44

*Ma se la pinza è montata "storta" ...*



Lezione 7 La programmazione

10 marzo 2014 45

*Esempio pratico:*

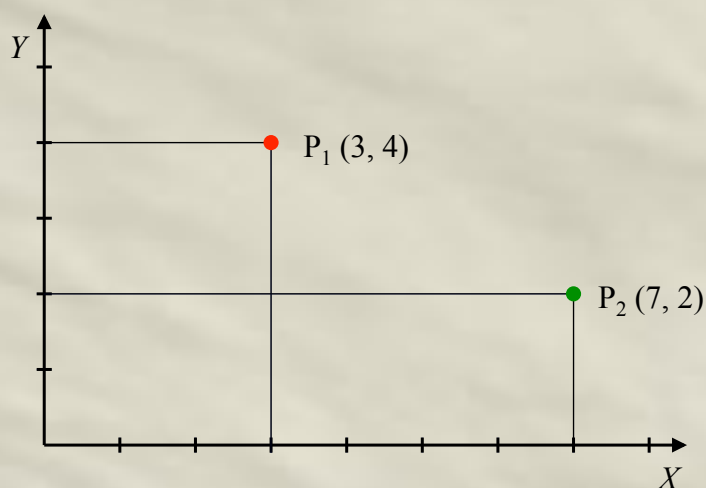


## *Facciamo questa considerazione:*

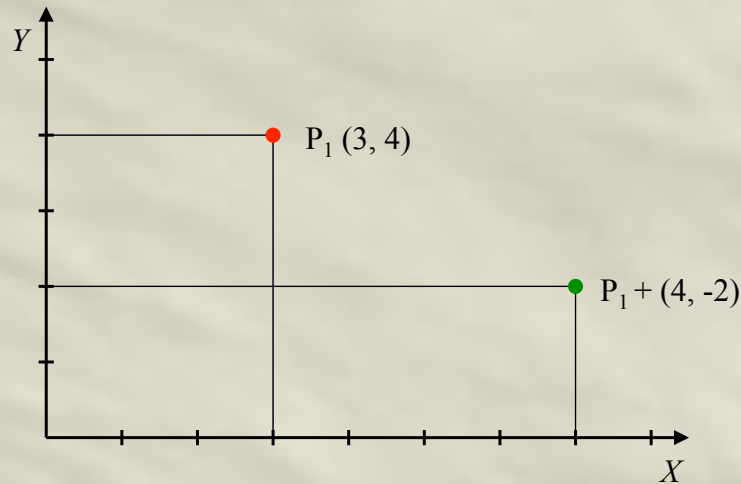
⇒ Una posa, espressa in coordinate cartesiane

- Non è altro che lo spostamento (rototraslazione) di una terna di assi (quella solidale all'end effector) dall'origine degli assi del robot
- Tant'è vero che si chiama anche *transformation*
- Ma uno spostamento può essere espresso in funzione di un punto qualunque...
- ...quindi anche di un'altra trasformazione

## *Ad esempio, nel piano:*





*Ad esempio, nel piano:*

Lezione 7 La programmazione

10 marzo 2014 49

*Una definizione:*

## ⇒ TCP (Tool Central Point):

- L'origine della terna di assi che viene assunta come posizione dell'end effector
- Per una pinza, in genere coincide con il punto centrale fra le ganasce
- Per una ventosa, con il centro della ventosa
- Per una matrice di ventose, con il centro della matrice
- ...
- Ed è orientato in modo che l'asse Z sia orientato nella direzione in cui deve muoversi l'end effector per fare la sua operazione

Lezione 7 La programmazione

10 marzo 2014 50

*Allora, per tornare alla pinza, noi possiamo:*

- ⇒ Definire la trasformazione che porta dalla base della pinza (che coincide con la flangia del polso) al TCP
- ⇒ Dirlo al robot usando l'istruzione TOOL

*La flangia del Kawasaki:*

