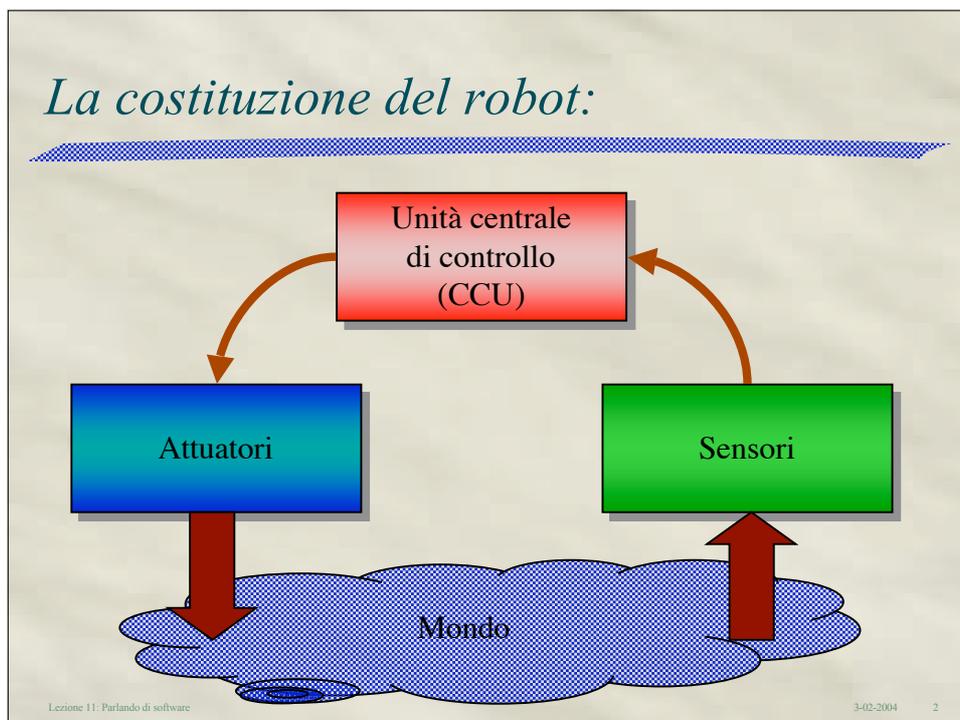


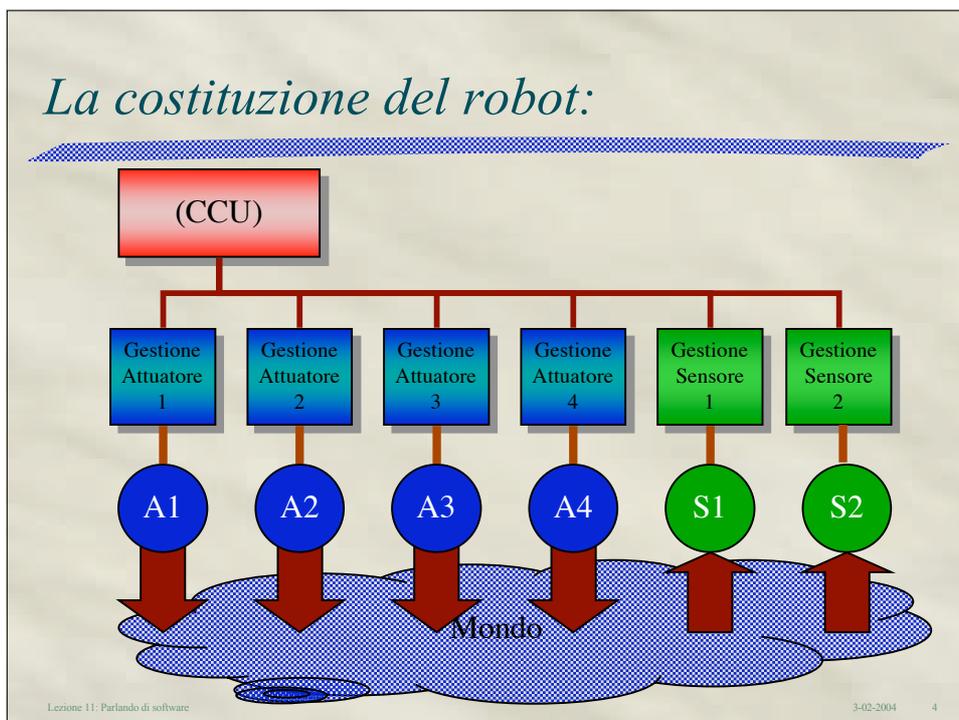
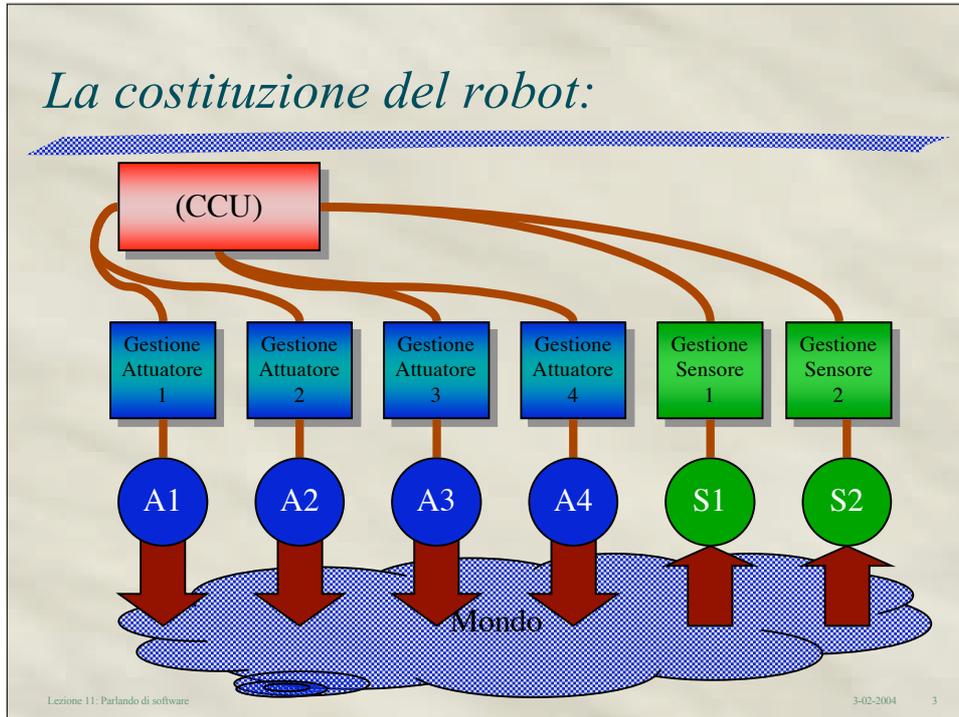
Robotica A

*Lezione 11:
Parlando di software*

La programmazione dei robot industriali

3-02-2004





Programmazione di un manipolatore industriale

⇒ 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

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.)

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 11: Parlando di software

3-02-2004 7

Ecco un "teach pendant"

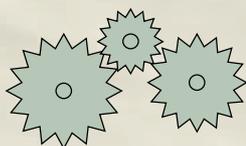


Lezione 11: Parlando di software

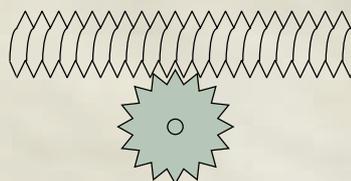
3-02-2004 8

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

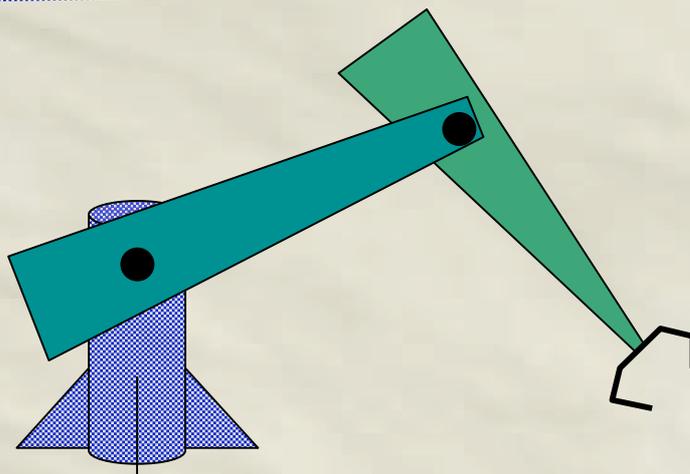


No!

Lezione 11: Parlando di software

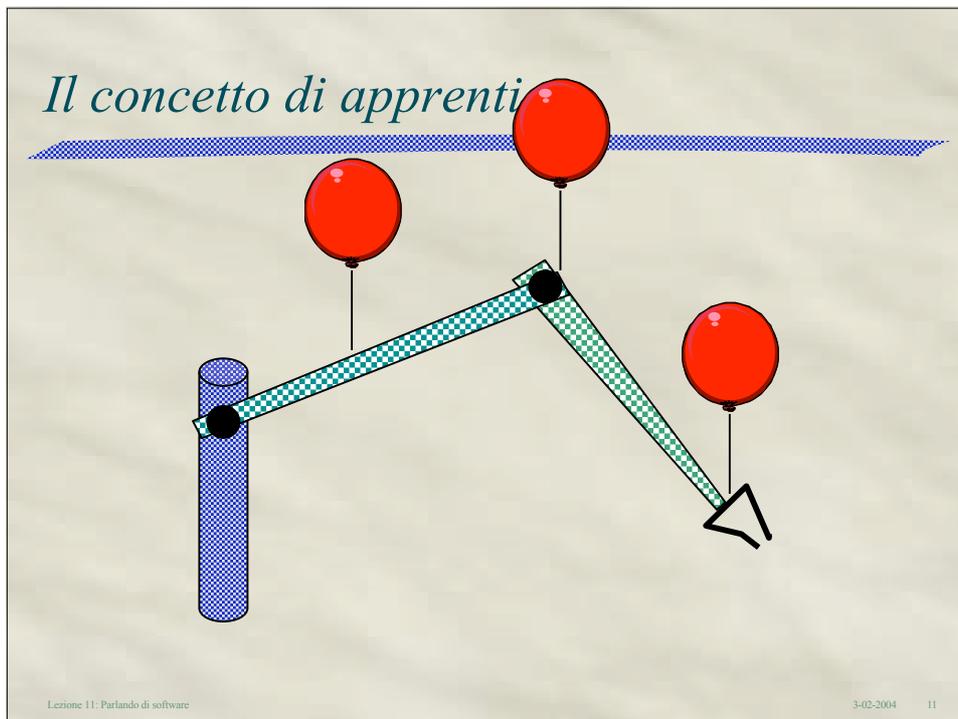
3-02-2004 9

Il concetto di apprendice



Lezione 11: Parlando di software

3-02-2004 10



*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 11: Parlando di software

3-02-2004 12

Un calcolatore con questi requisiti non esiste...

- ⇒ Però può essere emulato
- ⇒ Usando le classiche tecniche di emulazione già sperimentate per i calcolatori tradizionali
- ⇒ Attenzione alla differenza fra “simulatore” ed “emulatore”!
 - Emulate: To imitate the function of (another system), as by modifications to hardware or software that allow the imitating system to accept the same data, execute the same programs, and achieve the same results as the imitated system
 - Simulate: To create a representation or model of (a physical system or particular situation, for example).
- ⇒ Occorre definire una macchina virtuale che si comporti come il nostro calcolatore

Lezione 11: Parlando di software

3-02-2004 13

Quattro esempi di programmi per robot:

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

Lezione 11: Parlando di software

3-02-2004 14

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 11: Parlando di software

3-02-2004 15

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 11: Parlando di software

3-02-2004 16

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 11: Parlando di software

3-02-2004 17

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 11: Parlando di software

3-02-2004 18

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 11: Parlando di software

3-02-2004 19

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 11: Parlando di software

3-02-2004 20