

Esercizi con braccio e pinza (III)

```
c:\progra-1\mplab\asm1.asm
1:.....
2:;Programma di gestione della pinza di Pathfinder in modo controllato
3:
4:          LIST      p=16F87B          ;Tipo di processore
5:          include "16F87B.INC"        ;Definizione dei registri interni
6:
7:  TEMPO    EQU      0x70
8:
9:          ORG      0x00
10:
11:.....
12:;inizio del programma. Configurazioni
13:inizio    bcf      STATUS,RPD        ;Seleziona il banco 1
14:          movlw   0x07
15:          movwf   ANCONO             ;Porta digitale
16:          movlw   b'00000111'
17:          movwf   PORTB             ;Configurazione della PortaB
18:          movlw   0x07
19:          movwf   TMR0              ;Prescaler 256 per il timer
20:          bcf      STATUS,RPD        ;Seleziona il banco 0
21:          bcf      PORTB, 0          ;Pinza disattivata
22:
```

Realizzeremo un secondo esercizio di controllo della pinza di Pathfinder. In questo caso gestiremo la pinza nel modo controllato; attraverso questo metodo di funzionamento si chiuderà la pinza mediante un fronte di salita inviato sul pin RA3 del microcontroller, e grazie a un secondo fronte inviato sullo stesso pin, la pinza si aprirà. Come nel precedente esercizio utilizzeremo il finecorsa collegato su JP7 della scheda di interfaccia per controllare l'apertura e la chiusura della pinza.

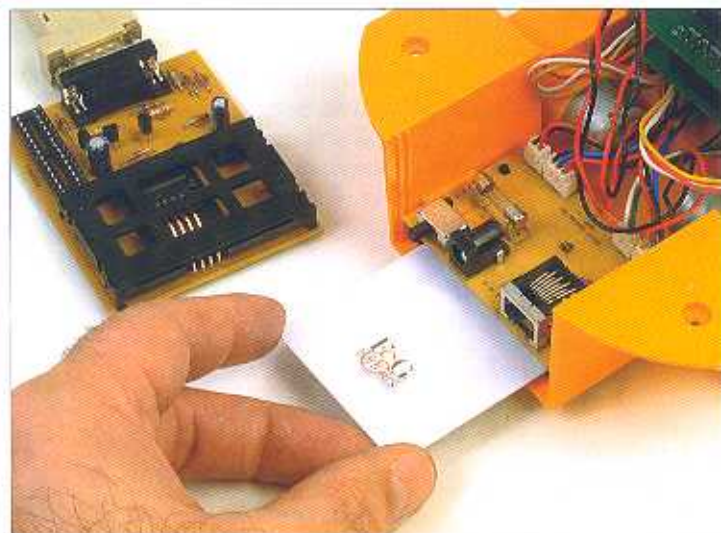
```
c:\progra-1\mplab\asm2.asm
22:.....
23:;Ciclo principale del programma
24:;Ciclo principale del programma
25:WAIT_FINECORSA: btfsz PORTA, 1
26:                goto  WAIT_FINECORSA
27:                bcf      PORTB, 0    ;Si invia un fronte che provocherà
28:                nap      PORTB, 0    ;l'apertura e la chiusura della pinza
29:                nap
30:                nap
31:                bcf      PORTB, 0
32:                call    DELAY        ;Si eliminano i rimbalzi del finecorsa
33:                goto    WAIT_FINECORSA
34:
```

Il ciclo principale del programma, verifica in continuazione lo stato del pin RA1 su cui è collegato il finecorsa. Ogni volta che si attiva il finecorsa; verrà inviato un fronte di salita sul pin RA3 del microcontroller. Se la pinza in quel momento è chiusa, il fronte di salita la farà aprire, e se è aperta l'impulso la farà chiudere. Il programma si ripete in modo infinito.

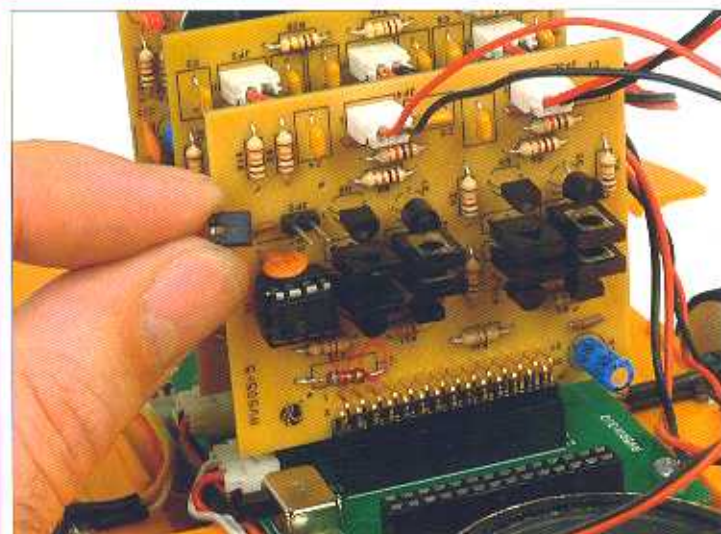
```
c:\progra-1\mplab\asm3.asm
35:.....
36:;Rutine di temporizzazione di ts
37:DELAY:        movlw   .100
38:              movwf   TEMPO
39:              movlw   0x01
40:              movwf   TMR0        ;temporizzazione da 100
41:              bcf      INTCOM, 2
42:              decfsz  TEMPO, 1
43:              goto    DEL
44:              decfsz  TEMPO, 1
45:              goto    DEL_10      ;Si ripete il ciclo 100 volte
46:              return
47:
```

Allo scopo di eliminare i rimbalzi del finecorsa, e lasciare il tempo necessario all'apertura e alla chiusura della pinza, fra gli impulsi del finecorsa è stata inserita una routine di temporizzazione di un secondo, controllata con il Timer 0 del microcontroller. Questa routine viene chiamata ogni volta che si attiva il finecorsa, per poter aspettare un secondo, prima di ritornare a testare lo stato del sensore.

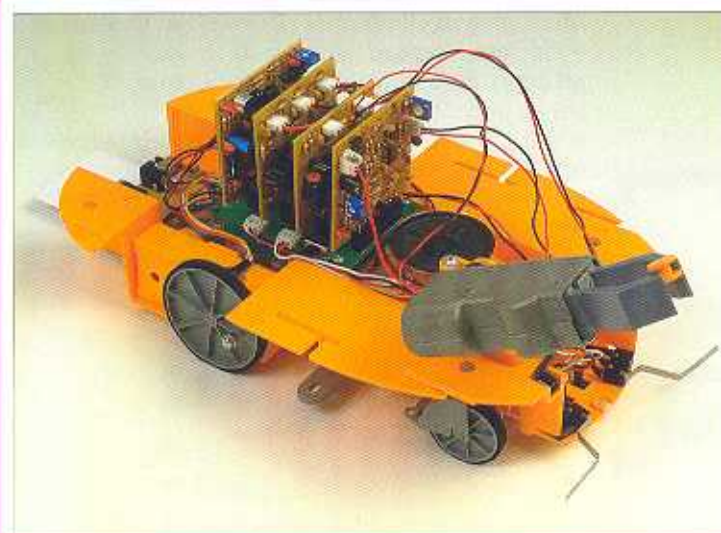
Esercizi con braccio e pinza (III)



Scriveremo l'esercizio con il programma MPLAB e lo compileremo per ottenere il file esadecimale. Utilizzeremo il software di scrittura ICPROG per trasferire il file alla Smartcard mediante la scheda di scrittura. Dopo aver programmato la Smartcard, la inseriremo sulla scheda di alimentazione di Pathfinder con l'orientamento adeguato.



Per fare in modo che la pinza del braccio di Pathfinder funzioni in modo controllato bisogna togliere il jumper JP2 della scheda di controllo del braccio e della pinza. Grazie a questo, il microcontroller PIC12C508 di questa scheda interpreterà i segnali inviati sul pin RA3 del microcontroller principale, come segnale di controllo sia dell'apertura che della chiusura della pinza. Per verificare l'esercizio, la scheda di controllo del braccio e della pinza deve essere inserita sul connettore JP15 della scheda di interfaccia e il motore della pinza collegato sul terminale JP4 di questa scheda.



Un'ulteriore verifica dell'esercizio, dovrà essere fatta alimentando il robot e attendendo qualche secondo in modo che il microcontroller possa leggere il programma dalla Smartcard. Grazie all'attivazione del finecorsa controlleremo sia l'apertura che la chiusura della pinza. Un primo impulso farà chiudere la pinza, la seconda attivazione farà aprire la pinza rilasciando l'oggetto che aveva in precedenza afferrato.

Esercizi con braccio e pinza (IV)

```
1 ;Programma di controllo del braccio e della pinza di Pathfinder. Il finecorsa collegato a RA1
2 ;preziona la salita e la discesa del braccio. Il finecorsa collegato su RA2 preziona
3 ;l'apertura e la chiusura della pinza
4
5 LIST p=16F478 ;Tipo di processore
6 include "16F478.INC" ;Definizione dei registri interni
7
8 ORG 0
9 TEMPO EQU 0x20
10 TEMPO2 EQU 0x22
11 TEN_BRACCIO EQU -5
12 TOFF_BRACCIO EQU -18
13 IMPULSO_ALZARE EQU -2
14 IMPULSO_ABBASSARE EQU -3
15
16 ORG 0x0000
17
```

Effettueremo ora un altro esercizio con il braccio di Pathfinder; in esso combineremo diversi esercizi che abbiamo già utilizzato in precedenza. Con questo programma controlleremo sia la salita che la discesa del braccio, e l'apertura e la chiusura della pinza. Utilizzeremo il finecorsa collegato su JP7 per il controllo del braccio, e il finecorsa su JP8 per il controllo dell'apertura e della chiusura della pinza. All'inizio del programma imposteremo i parametri di configurazione TON_BRACCIO, TOFF_BRACCIO, IMPULSO_ALZARE e IMPULSO_ABBASSARE, che compiono le stesse funzioni di controllo del motore del braccio dell'esercizio già realizzato arm1.asm.

```
18 ;Inizializza
19 ;Inizializza
20 movlw 0x07
21 movwf ACC0H
22 movlw 0x00001111
23 movwf PORTA
24 clrf TRISB ;Porta B si configura come uscita
25 movlw 0x00000001
26 movwf TMR0 ;Prescaler per il Timer0
27 bcf STATUS,RPB ;Selezione banca 0
28 movlw TON_BRACCIO
29 movwf TEMPO
30 clrf PORTB
31 movlw 0x00100001
32 movwf TMR1 ;SI spengono i motori
33 clrf TRIS1
34 bcf PORT1,TRIS1F
35 bcf PORTA,1
```

Questo programma utilizzerà i pin RA1 e RA2 come ingresso per finecorsa, e RA3 come uscita per il segnale di controllo della pinza. La porta B del microcontroller verrà configurata come uscita per la gestione dei motori. L'esercizio utilizza il Timer 0 e il Timer 1 per il controllo della modulazione dell'ampiezza degli impulsi dei motori e per i tempi di salita e di discesa del braccio del robot.

```
37 ;
38 ;
39 CICLO1: clrf PORTB ;Ciclo principale del programma
40 btfsc PORTA, 2 ;SI testa il finecorsa RA2
41 call PINZA ;per l'apertura/chiusura della pinza
42 btfsc PORTA, 1 ;SI attende il segnale su RA1
43 goto CICLO1 ;per alzare il braccio
44 movlw IMPULSO_ALZARE
45 movwf TEMPO2
46 clrf TRISB
47 bcf PORT1,TRIS1F
48 ;ATTESA_ALZARE_BRACCIO:
49 call BRACCIO_ALZARE ;SI alza il braccio del robot
50 btfsc PORT1,TRIS1F
51 goto ATTESA_ALZARE_BRACCIO
52 decfsz TEMPO2,1
53 goto ALZARE_BRACCIO
54 CICLO2: clrf PORTB
55 btfsc PORTA, 2 ;SI testa il finecorsa RA2
56 call PINZA ;per l'apertura/chiusura della pinza
57 btfsc PORTA, 1 ;SI attende il secondo segnale su RA1
58 goto CICLO2 ;per abbassare il braccio
59 movlw IMPULSO_ABBASSARE
60 movwf TEMPO2
61 btfsc PORT1,TRIS1F
62 clrf TRISB
63 bcf PORT1,TRIS1F
64 ;ATTESA_ABBASSARE_BRACCIO:
65 call BRACCIO_ABBASSARE ;SI abbassa il braccio del robot
66 btfsc PORT1,TRIS1F
67 goto ATTESA_ABBASSARE_BRACCIO
68 decfsz TEMPO2,1
69 goto ABBASSARE_BRACCIO
70 goto CICLO1
```

Questo è il ciclo principale del programma. Si testa lo stato dei segnali RA1 e RA2 che contengono lo stato dei finecorsa. Un '1' su RA2 richiamerà la routine, che ha il compito di inviare un impulso per RA3 per provocare l'apertura o la chiusura della pinza. L'attivazione del finecorsa RA1 fa iniziare il movimento di salita del braccio e una seconda attivazione di questo finecorsa provocherà la discesa del braccio.

Esercizi con braccio e pinza (IV)



```
c:\proga\lupia\lupia.asm
72:.....
73:;Restino di apertura e chiusura della pinza
74:PINZA:      bcf   PORTA, 3      ;! Isola un fronte che provocherà
75:            nop                    ;l'apertura e la chiusura della pinza
76:            nop
77:            nop
78:            nop
79:            bcf   PORTA, 3
80:            return
81:.....
```

La funzione che vediamo nell'immagine ha il compito di inviare un fronte di salita tramite il pin RA3 del microcontroller che provocherà l'apertura o la chiusura della pinza. Questo esercizio gestisce la pinza in modo controllato, dato che il jumper JP2 della scheda di controllo del braccio e della pinza dovrà essere tolto dalla sua posizione. Se il jumper rimane montato la pinza funzionerà in modo temporizzato, cioè un'attivazione del finecorsa RA2 provocherà un ciclo completo di apertura e chiusura.

```
c:\proga\lupia\lupia.asm
82:.....
83:;Funzione dedicata al movimento di salita del braccio
84:BRACCIO_ALZARE:
85:            btfss  IMCON, 2      ;overflow del timer
86:            return
87:            bcf   IMCON, 2
88:            decfsz TEMPO, 1
89:            return
90:            movf  PORTA, 0
91:            movlw b'11000000'
92:            movwf 0
93:            movlw 0
94:            subwf 0, 0
95:            btfss STATUS, 2
96:            goto  SPEDIRE_BRACCIO
97:            goto  ACCENDERE_BRACCIO
98:ACCENDERE_BRACCIO:
99:            bcf   PORTA, 7
100:           bcf   PORTA, 6
101:           movlw TON_BRACCIO
102:           movwf TEMPO
103:           return
104:SPEDIRE_BRACCIO:
105:           bcf   PORTA, 6
106:           bcf   PORTA, 7
107:           movlw TOFF_BRACCIO
108:           movwf TEMPO
109:           return
110:.....
```

Questa funzione ha il compito di controllare il motore del braccio nel movimento di salita. Possiamo modificare la velocità di salita e di discesa del braccio con le variabili che si trovano all'inizio del file. Se aumentiamo il valore della variabile TON_BRACCIO, il motore si muoverà più rapidamente e il braccio avrà più forza. Modificando il valore della variabile IMPULSO_ALZARE e IMPULSO_ABBASSARE controlliamo quanti secondi rimane attivato il motore del braccio sia nel movimento di salita che in quello di discesa.

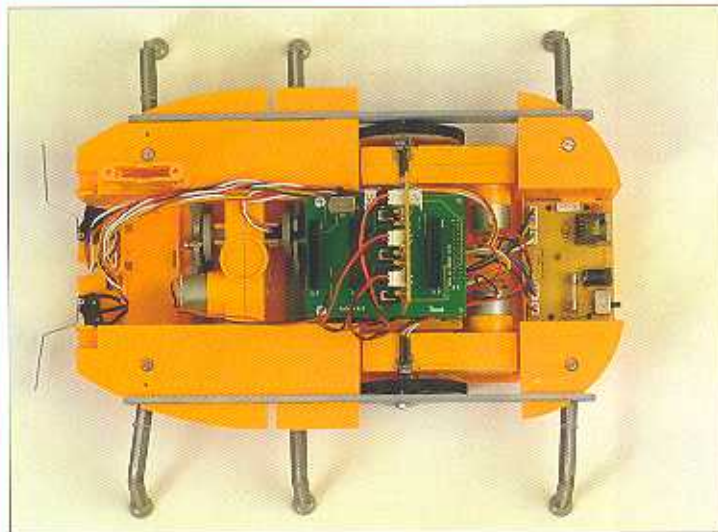
```
c:\proga\lupia\lupia.asm
111:.....
112:;Funzione dedicata al movimento di discesa del braccio
113:BRACCIO_ABBASSARE:
114:           btfss  IMCON, 2      ;overflow del timer
115:           return
116:           bcf   IMCON, 2
117:           decfsz TEMPO, 1
118:           return
119:           movf  PORTA, 0
120:           movlw b'11000000'
121:           movwf 0
122:           movlw 0
123:           subwf 0, 0
124:           btfss STATUS, 2
125:           goto  SPEDIRE_BRACCIO
126:           goto  ACCENDERE_BRACCIO
127:ACCENDERE_BRACCIO:
128:           bcf   PORTA, 7
129:           bcf   PORTA, 6
130:           movlw TON_BRACCIO
131:           movwf TEMPO
132:           return
133:SPEDIRE_BRACCIO:
134:           bcf   PORTA, 6
135:           bcf   PORTA, 7
136:           movlw TOFF_BRACCIO
137:           movwf TEMPO
138:           return
139:           END                    ;fine del programma sorgente
140:.....
```

Per provare l'esercizio, la scheda di controllo del braccio e della pinza di Pathfinder dovrà essere inserita sul connettore JP15 della scheda di interfaccia. Il motore del braccio sarà collegato sul connettore JP3 della scheda di controllo del braccio e il motore della pinza sul connettore JP4. Dopo avere scritto l'esercizio con MPLAB lo compileremo e lo scriveremo sulla Smartcard utilizzando il software ICPROG. Per verificare l'esercizio, attiveremo i finecorsa anteriori del robot, controllando il braccio e la pinza.

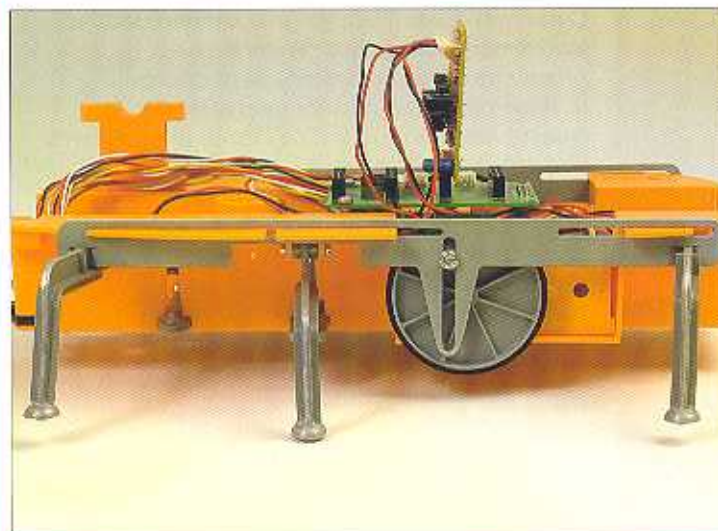
Meccanica di Pathfinder (III)



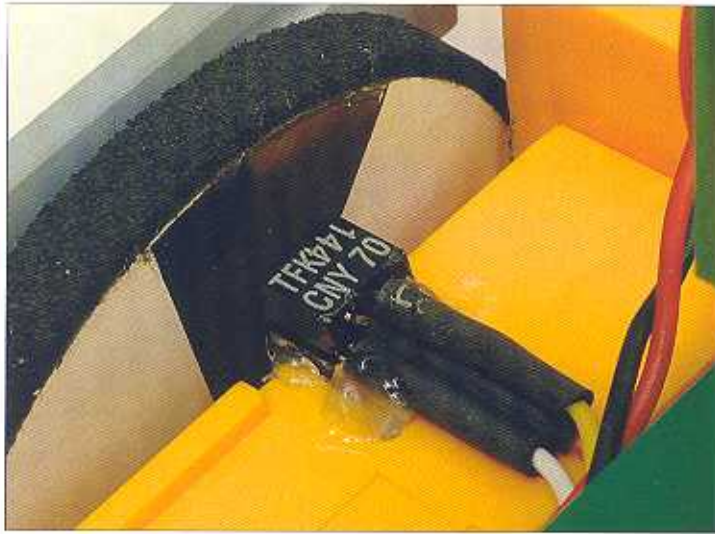
Abbiamo già analizzato la disposizione dei motori e i meccanismi di Pathfinder in configurazione veicolo. Spiegheremo ora il suo funzionamento nella configurazione con le zampe. Con questa configurazione Pathfinder disporrà di sei zampe mediante le quali potrà avanzare, retrocedere o girare in qualsiasi direzione. Il movimento sarà simile a quello che realizzano gli insetti con le loro zampe.



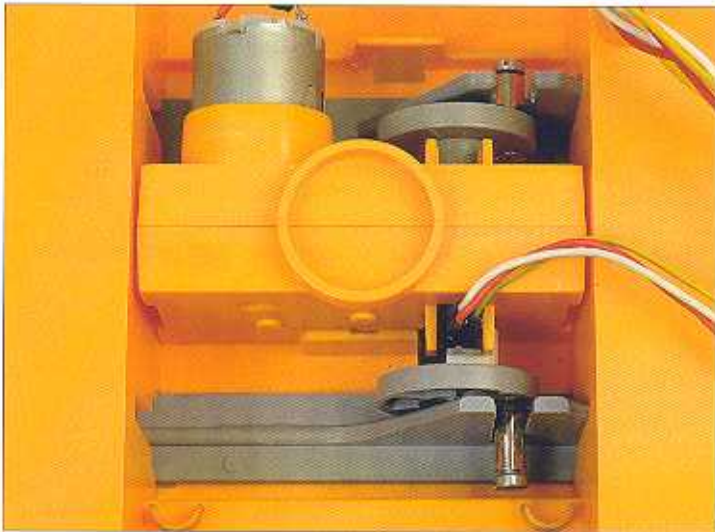
Le sei zampe di Pathfinder sono montate su tre meccanismi, ognuno dei quali muove due zampe. Mediante due meccanismi laterali vengono unite la zampa anteriore e quella posteriore del robot. Altri pezzi collocati al centro del robot, uniscono le due zampe centrali. Grazie a questi meccanismi le zampe posteriori e anteriori di entrambi i lati avanzano o retrocedono sempre in coppia.



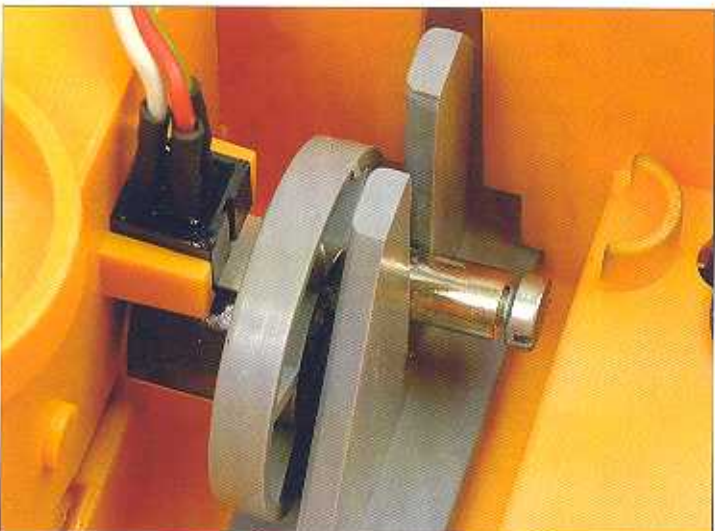
I due meccanismi laterali che uniscono una zampa anteriore e una posteriore, sono collegati alle ruote di Pathfinder tramite un bullone che verrà montato sulle stesse. Grazie a questo, il movimento di rotazione delle ruote, utilizzato in modo veicolo, verrà convertito in un movimento di traslazione in avanti e indietro di questo meccanismo, che servirà per muovere le zampe posteriori e anteriori.



Il segmento nero disegnato all'interno delle ruote, che è rilevato dai sensori di tipo ottico, servirà per controllare la massima estensione delle zampe che possiamo realizzare con il motore, quando Pathfinder funziona con questa configurazione. I limiti di questo settore nero, indicano i limiti massimi dei movimenti delle zampe del robot sia per lo spostamento in avanti che per la retromarcia.



Il meccanismo delle zampe centrali ha una funzione diversa. I meccanismi laterali servono a fare in modo che le zampe posteriori e anteriori possano avanzare e retrocedere, questo meccanismo centrale servirà a fare eseguire ai piedini centrali delle oscillazioni verso sinistra o verso destra. Grazie a questo movimento, il robot solleverà il suo lato sinistro o quello destro, e le zampe che rimarranno in aria potranno avanzare o retrocedere seguendo il movimento dei meccanismi laterali.

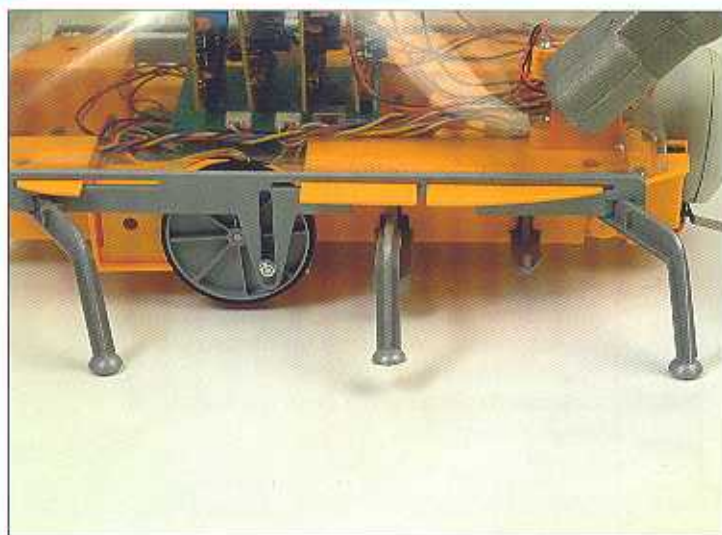


Come per il controllo dei meccanismi laterali, anche qui abbiamo un sensore ottico che rileva un settore nero collegato al movimento del motore centrale. Questo settore nero indica i limiti di movimento che dobbiamo applicare al motore centrale, per fare in modo che le zampe centrali si muovano verso destra o verso sinistra. In questo modo, per controllare il movimento di Pathfinder nella configurazione con le zampe, sarà indispensabile utilizzare tre sensori ottici, due per i piedini laterali e uno per quelli centrali.

Meccanica di Pathfinder (IV)



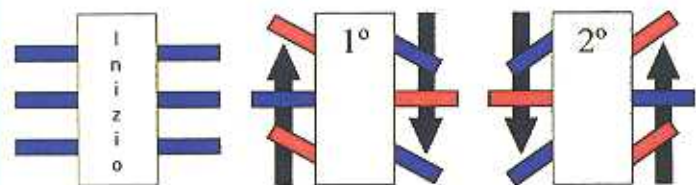
Continuiamo l'analisi della meccanica di Pathfinder nella configurazione con le zampe. Cercheremo di spiegare il principio di funzionamento dei motori laterali e centrali che Pathfinder utilizza per spostarsi. Quando Pathfinder si muove con le zampe, il suo avanzamento sarà più lento rispetto alle ruote, però dato che solleva le zampe mentre cammina, potrà percorrere terreni più accidentati, in cui un veicolo non potrebbe passare.



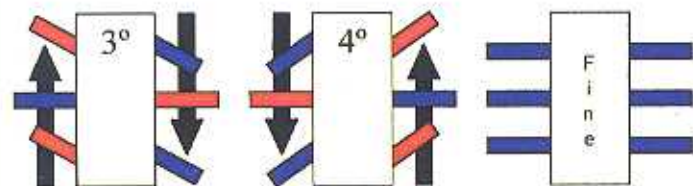
Per fare in modo che il robot possa avanzare con le zampe, dobbiamo capire che non potranno essere appoggiate al suolo tutte nello stesso tempo, altrimenti il robot trascinerrebbe le zampe e non potrebbe spostarsi. La chiave dei movimenti con le zampe consiste nel combinare quali rimangono sollevate e quali cambiano, realizzando i movimenti nell'ordine adeguato.



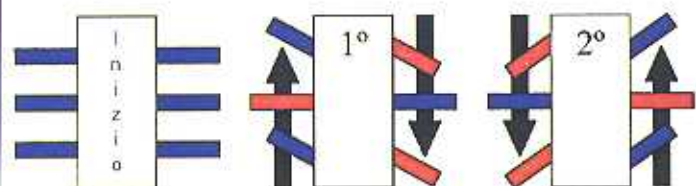
Le sei zampe del robot saranno tutte appoggiate solamente quando è fermo. Quando avanza, retrocede o gira, appoggerà sempre solamente tre zampe, per potersi muovere. Il meccanismo oscillante delle due zampe centrali sarà fondamentale, dato che servirà per inclinare il robot da entrambi i lati. In questo modo il robot appoggerà su un piede centrale e su due laterali lasciando il resto delle zampe sollevate dal suolo.



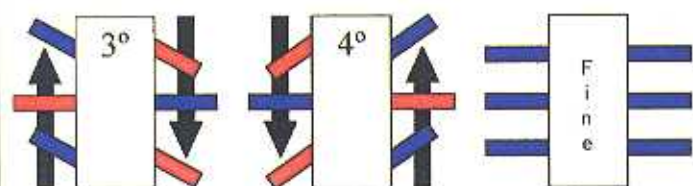
Marcia avanti



In questo grafico possiamo vedere la sequenza di movimento che realizza Pathfinder quando avanza nella configurazione con le zampe. Con il colore azzurro sono indicate le zampe appoggiate e con il colore rosso quelle che si trovano sollevate nelle varie sequenze del movimento. Combinando movimenti contrari con le zampe laterali appoggiate e quelle in aria, potremo fare realizzare al robot un movimento in avanti.



Marcia indietro

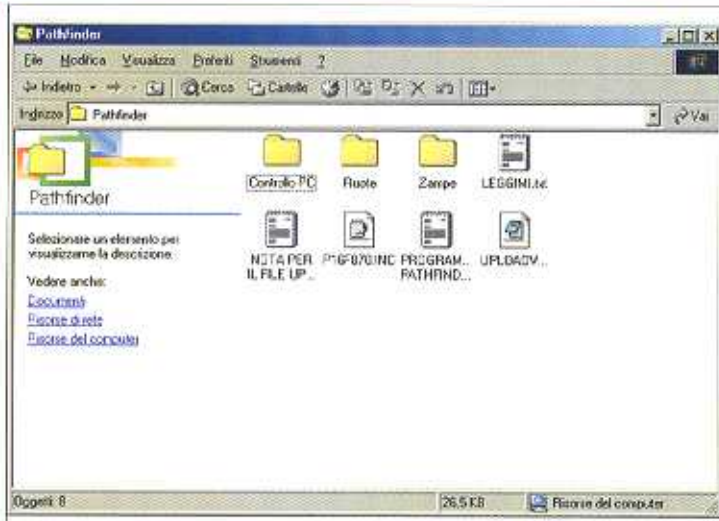


In questo secondo grafico, possiamo vedere la sequenza dei movimenti da applicare ai tre motori per fare in modo che Pathfinder realizzi un movimento di retromarcia nella configurazione con le zampe. La sequenza è molto simile a quella realizzata quando Pathfinder avanza, cambia solo il verso del movimento dei piedini laterali nei diversi appoggi. In questo caso avremo quattro sequenze diverse di movimento; a seconda se vogliamo fare avanzare il robot, farlo retrocedere, oppure farlo girare a destra o a sinistra.

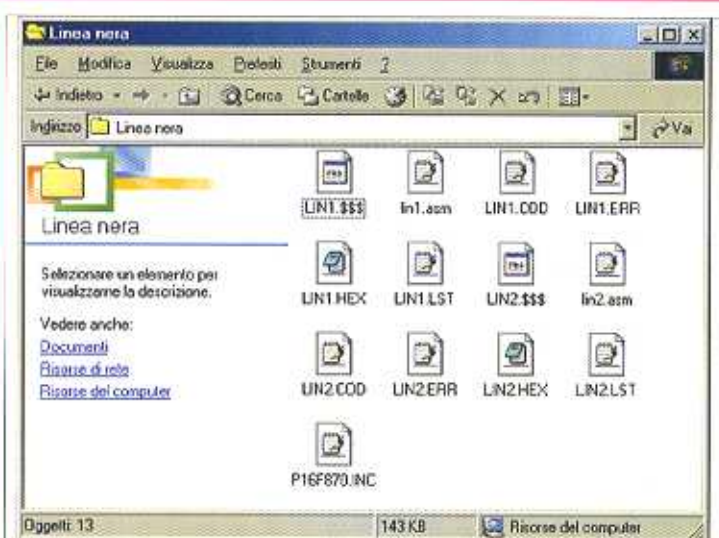


Quando Pathfinder funziona nella configurazione con le zampe, anche detta esapodo, non potremo utilizzare i sensori ottici anteriori, utili per seguire i percorsi nel modo veicolo, perché questi sensori rimarranno troppo sollevati rispetto al suolo e perché subiranno il movimento di oscillazione del robot. Tutti gli altri sensori, cioè quelli meccanici, a ultrasuoni, a infrarossi, audio, il braccio articolato, ecc. si possono utilizzare per far eseguire a Pathfinder qualsiasi applicazione tramite il programma.

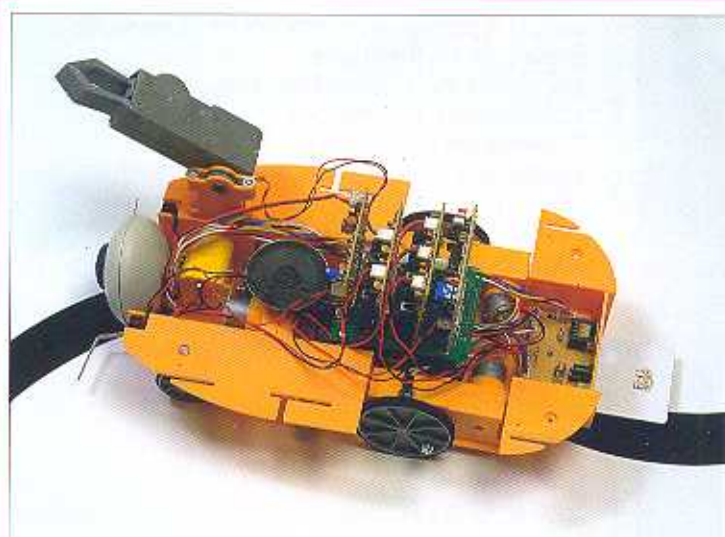
Modo ruote: Linea nera (I)



Sul secondo CD di quest'opera si trovano diversi esercizi per gestire Pathfinder nella configurazione con le ruote e nel modo esapodo, cioè nella configurazione con le zampe. Nella directory principale del CD troviamo tre cartelle, una chiamata Controllo PC che contiene i programmi per gestire il robot collegato direttamente a un computer, la cartella Zampe con gli esercizi per il controllo di Pathfinder in modo esapodo e la cartella Ruote, che contiene diversi esercizi per gestire Pathfinder in modo veicolo.



Inizieremo con gli esercizi di Pathfinder nella configurazione con le ruote, quindi faremo accesso alla cartella Ruote. All'interno di questa directory troveremo tre nuove cartelle, ognuna delle quali contiene tre diversi esercizi del robot in questa configurazione. Inizieremo con gli esercizi chiamati Linea Nera. In questi esercizi il robot potrà seguire un percorso segnato da una linea nera su un circuito che abbia un fondo di colore chiaro, idealmente bianco. A questo scopo utilizzerà i due sensori ottici posizionati nella parte anteriore del telaio del robot.



Per provare questi esercizi, da effettuare utilizzando una linea nera disegnata sul pavimento, il robot dovrà essere configurato in modo veicolo. In questo modo di funzionamento ognuno dei motori laterali avrà il compito di gestire una delle ruote di trazione. Il motore centrale del robot muoverà il meccanismo di direzione che servirà al robot per girare a sinistra e a destra, in base alle informazioni fornite dai sensori, per mantenersi costantemente sul percorso corretto.

