

## Modo ruote: Linea nera (III)

```
c:\spalm\lavoro\lavoro\linea\lin2.asm
1: *****
2: PROGRAMMA PER IL CONTROLLO DEL ROBOT PATHFINDER IN MODO RUOTE. Il programma segue il t
3: su di un fondo bianco. Se si rileva un contatto tramite i finecorsa, il robot si ferma
4:
5:
6: R00 e R01 sono per il motore centrale
7: R02: 1 -> Robot gira verso destra
8: R02 e R03 sono per il motore della ruota sinistra
9: R02: 0
10: R02: 1 -> Robot avanza
11: R04 e R05 sono per il motore della ruota destra
12: R04: 1
13: R05: 0 -> Robot avanza
14:
15:
16:
17: Quando i sensori rilevano nero inviano un '1'. Se rilevano bianco inviano uno '0'
18: RC0: Sensore di sinistra (JP12)
19: RC1: Sensore di destra (JP17)
20: RC2: Sensore centrale (JP18)
21: *****
```

Nel fascicolo precedente abbiamo analizzato il funzionamento del programma lin1.asm, che permette al robot di seguire un percorso indicato mediante una linea nera su uno sfondo bianco. Analizzeremo ora il programma lin2.asm, anch'esso segue una linea nera compiendo le stesse funzioni del programma precedente, però è stata aggiunta la rilevazione degli ostacoli mediante finecorsa. Se questi si attivano il robot si fermerà.

```
c:\spalm\lavoro\lavoro\linea\lin2.asm
22
23 list p=16f870 ;Definizione del tipo di dispositivo
24 include "16f870.inc" ;Libreria con i registri di controllo
25
26 Variabili del programma che saranno utilizzate come registri (memoria 800 del PIC)
27 TONR0 equ 0x20 ;Variabile ausiliaria per la modulazione di ampiezza degli impulsi con il timer
28 ONR0 equ 0x21 ;Variabile ausiliaria per operazioni aritmetiche
29 SX equ 0x22 ;Variabile che si imposta a 1 quando il robot muove verso sinistra. Si è attivato di
30 SX equ 0x23 ;Variabile che si imposta a 1 quando il robot muove verso destra. Si è attivato di
31 AUSILIARIO SINISTRA EQU 0x00 ;Variabile ausiliaria per evitare il rumore prodotto dal sensore sinistra che gi
32 AUSILIARIO DESTRA EQU 0x01 ;Variabile ausiliaria per evitare il rumore prodotto dal sensore destra che gi
33
34 Definizione di alcuni parametri del programma
35 TON_AVANZA equ 5
36 TOFF_AVANZA equ 8
37 TON_CENTRALE equ 5
38 TOFF_CENTRALE equ 10
39 CONDIZIONE_STARTABLE EQU 10
40
41 org 0x00
```

All'inizio del programma ci sono diverse variabili che servono per controllare la velocità di rotazione dei motori del robot. Le variabili TON\_AVANZA e TOFF\_AVANZA controllano la velocità di avanzamento di Pathfinder. In funzione dello stato delle batterie, potrebbe essere necessario modificare questi valori. Più alto sarà il rapporto del valore TON\_AVANZA in relazione a TOFF\_AVANZA, maggiore sarà la velocità delle ruote posteriori. Allo stesso modo le variabili TON\_CENTRALE e TOFF\_CENTRALE controllano la modulazione di ampiezza di impulsi applicati al motore centrale del robot.

```
c:\spalm\lavoro\lavoro\linea\lin2.asm
42 *****
43 INIZIO *****
44
45
46
47 *****
48
49 STATUS, 5 ;Banca 1
50 movlw 0x07
51 movwf ADCON0 ;PortA come ingresso digitale
52 movlw 0x00001111
53 movwf PORTA ;RA1 e RA2 ingressi per i finecorsa
54 cwf PORTB ;PortB come uscita per i sensori
55 movlw 0x00001111
56 movwf PORTC
57 movlw 0x00000001
58 movwf TMR0 ;prescaler. Mediante esso si sceglie la frequenz
59 bcf STATUS, 5 ;Banca 0
60 bcf INCON, 7 ;Si disabilitano gli interrupt
61 cwf PORTB
62 movlw TOFF_AVANZA
63 movwf TMR2
64 movlw TOFF_CENTRALE
65 movwf TMR2E2
66 movlw 10
67 movwf AUSILIARIO_DESTRA
68 movlw 10
69 movwf AUSILIARIO_SINISTRA
70 cwf 0x
71 cwf 0x
72 call DELAY
73
74
75 *****
76 POSIZIONE MOTORI CENTRALE:
77 bcfsc PORTC, 2
78 goto ATTENDI_P1ST0
79 call AVANZA_MOTORE_CENTRALE
80 bcf PORTB, 0 ;Si muove un po' il motore centrale
81 bcf PORTB, 1
82 call DELAY
83
84 *****
```

Le configurazioni realizzate all'inizio del programma sono simili a quelle effettuate in lin1.asm. Ora è stata aggiunta la configurazione della porta A come ingresso digitale, e i segnali RA1 e RA2 come ingressi per ricevere i segnali dei finecorsa. Per fare in modo che questi inviino i segnali al microcontroller, devono essere correttamente collegati sulla scheda di interfaccia e con il commutatore di questa scheda nella posizione relativa ai finecorsa.

## Modo ruote: Linea nera (III)

```
c:\path1\1vedolo\1ruote\lin2\lin2.asm
82 :*****
83 :Ciclo principale del programma
84 ATTENDI_PISTA:
85   clrfr PORTD
86   btfss PORTC, 0
87   goto  ATTENDI_PISTA
88   btfsc PORTC, 1
89   goto  CICLO
90   goto  ATTENDI_PISTA
91 :Sino a che il robot non è correttamente posizionato sulla pista non parte il pr
92 :Questo succede quando i due sensori isolano un '1'
93 CICLO:
94   bcf PORTB, 0 ;Si ferma il motore centrale
95   bcf PORTB, 1
96   btfsc PORTA, 1 ;Si testa lo stato del finecorsa
97   goto  FINECORSO_ON
98   btfsc PORTA, 2
99   goto  FINECORSO_ON
100  call  MOTORI_TRAZIONE
101  movf  PORTC, 0
102  andlw b'00000011'
103  xorlw b'00000011'
104  btfsc STATUS, 2
105  goto  AVANZARE
106  movf  PORTC, 0
107  andlw b'00000011'
108  xorlw b'00000011'
109  btfsc STATUS, 2
110  goto  CIRARE_DESTRA
111  movf  PORTC, 0
112  andlw b'00000011'
113  xorlw b'00000011'
114  btfsc STATUS, 2
115  goto  CIRARE_SINISTRA
116  goto  CICLO
```

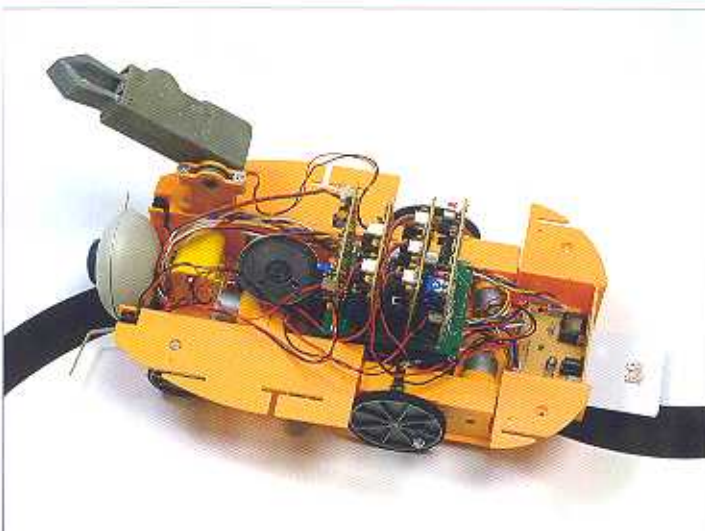
Il ciclo principale del programma segue gli stessi principi di lin1.asm. Si testano i segnali dei sensori ottici e in funzione del loro stato, si realizza la correzione corrispondente sui motori.

Nella routine è compreso il test dei nuovi segnali dei due pin RA1 e RA2, corrispondenti ai finecorsa. Se uno dei due segnali si attiva il robot si dovrà fermare, perché avrà toccato qualche ostacolo sul suo percorso.

```
c:\path1\1vedolo\1ruote\lin2\lin2.asm
186 FINE_AVANZAMENTO:
187   clrfr SX
188   clrfr DX
189   movlw .10
190   movwf AUXILIARIO_DESTRA
191   movlw .10
192   movwf AUXILIARIO_SINISTRA
193   goto  CICLO
194
195 :*****
196 :A questa routine si arriva se qualche finecorsa si è attivato, il robot è entra
197 FINECORSO_ON:
198   clrfr PORTB ;Si fermano i motori
199   goto  CICLO
200
201 :*****
202 :Funzione che gestisce l'avanzamento dei motori posteriori. Tramite questa routi
203 :si controlla la velocità di avanzamento del robot mediante modulazione di arpie
204 MOTORI_TRAZIONE:
205 :Controllo della modulazione di ampiezza degli impulsi per i motori di avvanza
206   btfss INTCON, 2 ;Overflow del timer
207   return
208   bcf INTCON, 2
209 MOTORI_TRAZIONE_AUX:
210   decfsz TEMPO, 1
211   return
```

La funzione da eseguire, quando i finecorsa si attivano, si chiama FINECORSO\_ON e a essa si accede mediante l'attivazione di uno qualsiasi dei due sensori.

In questa routine fermeremo il funzionamento di tutti i motori del robot fino a quando il finecorsa è attivo. Quando il finecorsa cessa di segnalare la presenza di un ostacolo, il robot riprenderà la sua marcia sul circuito.



In questa immagine osserviamo Pathfinder su un percorso del tipo linea nera, che dobbiamo costruire per provare gli esercizi. Possiamo utilizzare del nastro isolante di colore nero su una superficie riflettente, idealmente di colore bianco. Programmeremo la Smartcard con i programmi lin1.hex e lin2.hex, la inseriremo nella scheda di alimentazione di Pathfinder e attenderemo qualche secondo per dare modo al robot di leggere il contenuto del programma e iniziare l'esecuzione.