

IL GUERRIERO ROBOZAK

Analizziamo un esempio di codice RoboBasic con il quale RoboZak può compiere delle mosse di arti marziali.

Dopo avere dedicato gli ultimi fascicoli di RoboZak all'analisi del programma **Template.bas**, vediamo nelle prossime pagine un nuovo esempio di programmazione in **RoboBasic** per l'umanoide robotico (il file

dell'esempio, **ArtiMarziali.bas**, è presente all'interno del **sesto CD-Rom**). Grazie al codice il robot esegue una serie di sequenze motorie ispirate alle arti marziali. Prima di caricare il programma sulla scheda di RoboZak, è necessario caricare



COMPONENTI

◀1▶ **Motore elettrico del diciassettesimo servomotore (tipo 5N400 - cavo 40 cm)**



su quest'ultima il programma **Template.bas**. Nel nuovo esempio, infatti, si ricorre più volte all'istruzione **ACTION**, che permette di invocare alcune routine di movimento presenti nel codice **Template.bas**, in base al principio della multiprogrammazione (fascicolo 81, pagine 10, 11 e 12). Qualora nell'esecuzione delle sequenze motorie il robot perda l'equilibrio, verifica che gli **zero point** dei servomotori siano settati correttamente (per il settaggio di questi ultimi, fai riferimento a quanto spiegato nel fascicolo 75, pagine 11 e 12, oppure al video tutorial contenuto nel sesto **CD-Rom**).

CODICE ROBOBASIC: ARTIMARZIALI.BAS>>>

Come già accennato a pagina 7 di questo fascicolo, prima di caricare il seguente esempio sulla scheda di controllo di RoboZak è necessario caricare (o avere già caricato) il programma **Template.bas**, in modo che l'esempio **ArtiMarziali.bas** risieda nella parte bassa della memoria, mentre l'esempio **Template.bas** in quella alta. In questo modo è possibile sfruttare la **multiprogrammazione**: infatti nel programma seguente sono presenti alcune istruzioni di tipo **ACTION**, che richiamano le routine di movimento del programma **Template.bas**. Per ulteriori dettagli relativi alla modalità **multiprogrammazione** e al comando **ACTION**, fai riferimento al fascicolo 81 (pagine 10, 11 e 12).

```

\===== ACTION 20 ` right_attack
` ArtiMarziali.bas DELAY 2000
\===== GOSUB right_kick
DELAY 1000
PTP SETON ACTION 14 ` left_shift
PTP ALLON DELAY 1000
ACTION 15 ` left_attack
DELAY 2000
\== motor direction setting
DIR G6A,1,0,0,1,0,0 GOSUB left_kick
DIR G6B,1,1,1,1,1,1 DELAY 2000
DIR G6C,0,0,0,0,0,0 GOSUB meditation_pose
DIR G6D,0,1,1,0,1,0 GOSUB LED_OFF
END

\== motor start position read
GETMOTORSET G6A,1,1,1,1,1,0 standard_pose:
GETMOTORSET G6B,1,1,1,0,0,0 MOVE G6A,100,76,145,93,100,100
GETMOTORSET G6C,1,1,1,0,0,0 MOVE G6D,100,76,145,93,100,100
GETMOTORSET G6D,1,1,1,1,1,0 MOVE G6B,100,30,80,100,100,100
MOVE G6C,100,30,80,100,100,100
WAIT
RETURN

\== motor power on
SPEED 5
MOTOR G24

GOSUB LED_OFF meditation_pose:
GOSUB meditation_pose SPEED 3
DELAY 2000 MOVE G6A,100,151,23,140,100,100
TEMPO 230 MOVE G6B,100,30,80,100,100,100
GOSUB LED_ON MOVE G6C,100,30,80,100,100,100
MUSIC "ABCABC" MOVE G6D,100,151,23,140,100,100
GOSUB standard_pose MOVE G8A,101,154,22,157,100,100
DELAY 1000 MOVE G6B,101,30,80,100,100,100
MOVE G6C,101,32,82,100,100,100
MOVE G6D,100,156,23,154,97,100
DELAY 500
ACTION 11 ` forward_walk MOVE G8A,101,156,21,157,100,100
DELAY 1000 MOVE G6B,146,30,80,100,100,100
ACTION 1 ` bow MOVE G6C,144,34,84,100,100,100
DELAY 1000 MOVE G6D,101,156,23,155,95,100
ACTION 13 ` right_shift
DELAY 1000

```

```

DELAY 500
MOVE G6A,101,159,21,156,102,100
MOVE G6B,151,15,41,100,100,100
MOVE G6C,148,10,37,100,100,100
MOVE G6D,101,160,23,155,94,100
DELAY 3000
MOVE G6A,101,156,21,157,100,100
MOVE G6B,146,30,80,100,100,100
MOVE G6C,144,34,84,100,100,100
MOVE G6D,101,156,23,155,95,100
DELAY 500
MOVE G6A,100,151,23,140,100,100
MOVE G6B,100,30,80,100,100,100
MOVE G6C,100,30,80,100,100,100
MOVE G6D,100,151,23,140,100,100
RETURN

left_kick:
SPEED 3
MOVE G6A,100,76,125,125,100,100
MOVE G6D,100,76,125,125,100,100
WAIT
MOVE G6A,100,145,32,140,100,100
MOVE G6D,100,145,32,140,100,100
MOVE G6B,140,30,80,100,100,100
MOVE G6B,170,20,20,100,100,100
MOVE G6C,170,30,80,100,100,100
MOVE G6D,110,145,32,140,100,100
MOVE G6A,100,145,32,140,109,100
WAIT
MOVE G6D,110,145,32,140,104,100
WAIT
MOVE G6D,115,145,32,140,100,100
MOVE G6A,100,145,32,140,140,100
WAIT
MOVE G6D,115,145,32,140,122,100
DELAY 1000
SPEED 15
HIGH SPEED SETON
MOVE G6A,93,83,136,103,152,100
MOVE G6D,115,145,32,140,140,100
DELAY 1000
HIGH SPEED SETOFF
SPEED 5
MOVE G6A,93,144,52,103,152,100
MOVE G6A,93,144,52,103,89,100
MOVE G6D,115,145,32,158,117,100
MOVE G6A,93,144,52,120,89,100
MOVE G6D,97,134,55,129,112,100

right_kick:
SPEED 3
MOVE G6A,100,76,125,125,100,100
MOVE G6D,100,76,125,125,100,100
WAIT
MOVE G6D,100,145,32,140,100,100
MOVE G6A,100,145,32,140,100,100
MOVE G6C,140,30,80,100,100,100
MOVE G6C,170,20,20,100,100,100
MOVE G6B,170,30,80,100,100,100
MOVE G6A,110,145,32,140,100,100
MOVE G6D,100,145,32,140,109,100
WAIT
MOVE G6A,110,145,32,140,104,100
WAIT
MOVE G6A,115,145,32,140,100,100
MOVE G6D,100,145,32,140,140,100
WAIT
MOVE G6A,115,145,32,150,122,100
DELAY 1000
SPEED 15
HIGH SPEED SETON
MOVE G6D,93,83,136,103,152,100
MOVE G6A,115,145,32,150,140,100
DELAY 1000
HIGH SPEED SETOFF
SPEED 5
MOVE G6D,93,144,52,103,152,100
MOVE G6D,93,144,52,103,89,100
MOVE G6A,115,145,32,158,117,100
MOVE G6D,93,144,52,120,89,100
MOVE G6A,97,134,55,129,112,100
MOVE G6C,170,61,75,100,100,100
GOSUB standard_pose
RETURN

LED_OFF:
OUT 52,1
RETURN

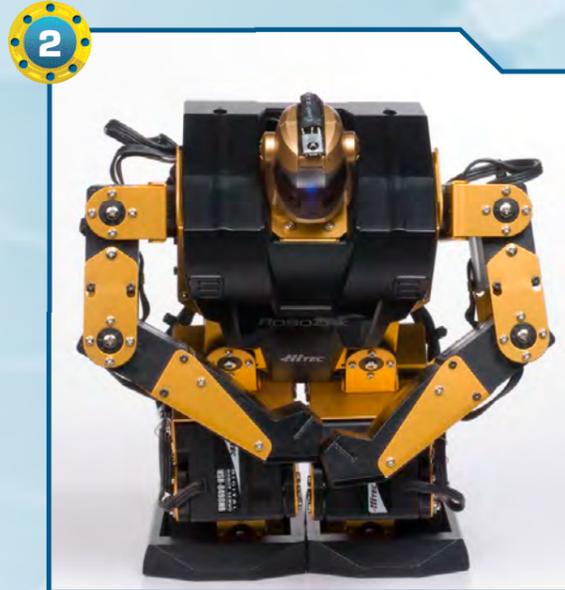
LED_ON:
OUT 52,0
RETURN

MOVE G6B,170,61,75,100,100,100
GOSUB standard_pose
RETURN

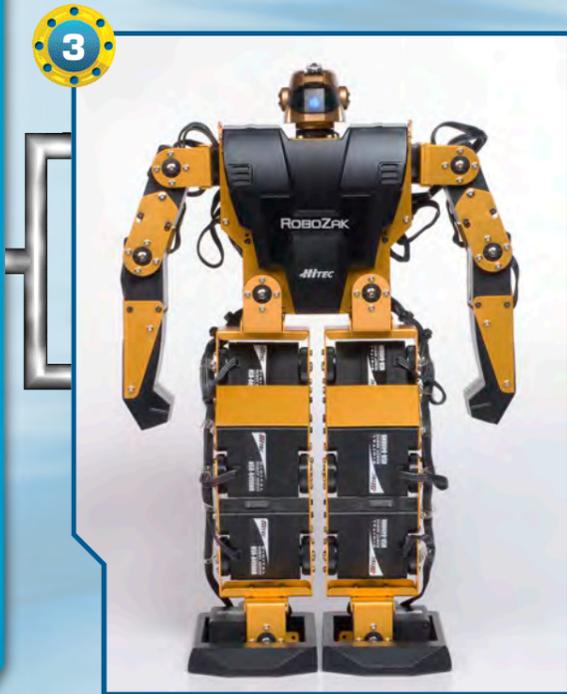
```



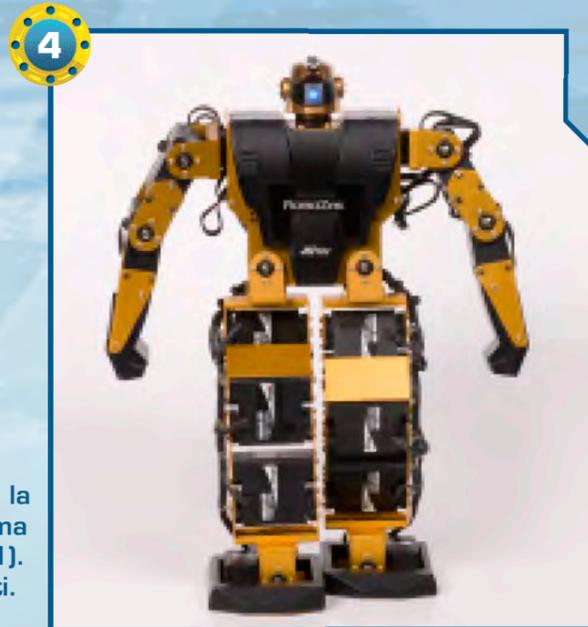
◀1▶ Inizialmente RoboZak si porta nella posizione 'meditation_pose', con il diodo LED spento.



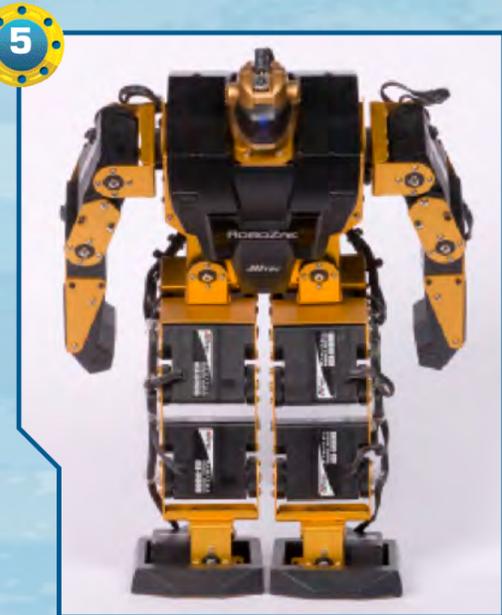
◀2▶ Dopo due secondi il LED si illumina e il robot emette una breve sequenza musicale.



◀3▶ RoboZak si porta in posizione standard (standard_pose).



◀4▶ Dopo un secondo viene invocata la routine 'forward_walk' del programma 'template.bas' (comando ACTION 11). Il robot compie alcuni passi in avanti.



◀5▶ Dopo un secondo viene invocata la routine 'bow' del programma 'template.bas' (comando ACTION 1). RoboZak effettua un inchino frontale.



◀6▶ Dopo un secondo viene invocata la routine 'right_shift' del programma 'template.bas' (comando ACTION 13). RoboZak effettua un passo laterale verso destra.



◀7▶ Dopo un secondo viene invocata la routine 'right_attack' del programma 'template.bas' (comando ACTION 20). RoboZak effettua un attacco laterale a destra.



◀8▶ Dopo due secondi RoboZak esegue un calcio acrobatico con la gamba destra (routine 'right_kick').

9



◀9▶ Dopo un secondo viene invocata la routine 'left_shift' del programma 'template.bas' (comando ACTION 14). RoboZak effettua un passo laterale verso sinistra.

10



◀10▶ Dopo un secondo viene invocata la routine 'left_attack' del programma 'template.bas' (comando ACTION 15). RoboZak effettua un attacco laterale a sinistra.

11



◀11▶ Dopo due secondi RoboZak esegue un calcio acrobatico con la gamba sinistra (routine 'left_kick').

12



◀12▶ Dopo due secondi il LED si spegne e RoboZak si riporta nella posizione di partenza ('meditation_pose').

RIEPILOGO COMPONENTI

In questo elenco trovi tutte le tipologie di pezzi che ti sono state fornite a partire dal primo fascicolo: puoi consultarlo quando devi affrontare le fasi di montaggio, in modo da avere un riferimento immediato per i componenti che dovrai utilizzare e per quelli che hai a disposizione.

- ▶ armatura del dorso
- ▶ armatura del torace
- ▶ base inferiore per servo A
- ▶ base inferiore per servo B
- ▶ base inferiore per servo C
- ▶ base superiore per servo A
- ▶ base superiore per servo B
- ▶ base superiore per servo C
- ▶ bullone da 3x4 mm
- ▶ caricabatterie
- ▶ cavo di prolunga per pacco batterie
- ▶ cavo seriale
- ▶ circuito con LED
- ▶ coperchio vano batterie
- ▶ copertura in plastica del piede sinistro e destro
- ▶ cuscinetto a sfera
- ▶ distanziatore da 3x5 mm
- ▶ elementi plastici della mano
- ▶ fascetta di fissaggio dei cavi
- ▶ fascetta di metallo
- ▶ fascetta in plastica per il raggruppamento dei cavi
- ▶ guaina in plastica proteggi cavo
- ▶ intelaiatura metallica del dorso
- ▶ intelaiatura metallica del piede
- ▶ intelaiatura metallica superiore
- ▶ intelaiatura metallica del polso
- ▶ intelaiatura metallica del torace
- ▶ motore elettrico cavo 200 mm (6N200 - Servo C)
- ▶ motore elettrico cavo 300 mm (4N300 - Servo A)
- ▶ motore elettrico cavo 400 mm (5N400 - Servo B)
- ▶ nastro biadesivo
- ▶ pacco batterie ricaricabili
- ▶ parte anteriore della testa
- ▶ parte posteriore della testa
- ▶ perno da 1,6x14 mm
- ▶ perno da 1,6x9 mm
- ▶ protezione per scheda MR-C3024
- ▶ ricevitore IR
- ▶ rondella da 6x2,2x0,5 mm
- ▶ rondella da 7,6x2,8x0,5 mm
- ▶ ruota dentata di tipo 1
- ▶ ruota dentata di tipo 2
- ▶ ruota dentata di tipo 3
- ▶ ruota dentata di tipo 4
- ▶ scheda MR-C3024
- ▶ scheda PC Servo Control
- ▶ sensore di contatto
- ▶ sensore di distanza
- ▶ sensore di luce
- ▶ sensore di suono
- ▶ sostegno per potenziometro
- ▶ squadrette circolari per servo (tipo 1, 2, 3, 4)
- ▶ squadretta circolare per il fissaggio della testa
- ▶ squadretta metallica a I
- ▶ squadrette metalliche a U (16 fori e 22 fori)
- ▶ squadretta metallica ad H
- ▶ squadretta metallica spalle (interna ed esterna)
- ▶ telecomando Remocon
- ▶ tubetto di grasso
- ▶ visiera
- ▶ vite di tipo M da 2,6x4 mm
- ▶ vite di tipo M da 2x4 mm
- ▶ vite di tipo M da 3x4 mm
- ▶ vite di tipo T-2 da 2,6x6 mm
- ▶ vite di tipo T-2 da 2x12 mm
- ▶ vite di tipo T-2 da 2x18 mm
- ▶ vite di tipo T-2 da 2x21 mm (nera)
- ▶ vite di tipo T-2 da 2x26 mm (nera)
- ▶ vite di tipo T-2 da 2x4 mm
- ▶ vite di tipo T-2 da 2x5 mm
- ▶ vite di tipo T-2 da 2x8 mm

