

Istruzioni che non coinvolgono il movimento nel linguaggio MIKAN I

Nei capitoli precedenti abbiamo studiato le cinque istruzioni di movimento: MOV, ROT, CIE, ABR e VEL del linguaggio MIKAN I per la gestione del robot Armdroid, vediamo ora quelle rimanenti.

PUN (PUNTATORE)

Si tratta di un'istruzione non operativa. La sua esecuzione serve per registrare nella memoria la posizione dell'istruzione successiva da eseguire. Non ha operandi.

FIN (FINALE)

È un'altra istruzione non operativa, che si utilizza per indicare al sistema di controllo il termine dell'esecuzione di un programma. Il suo utilizzo è obbligatorio al termine di ogni programma. È possibile inserirla in qualsiasi punto di un programma per provocarne l'arresto. Non ha operandi.

SAL (SALTO)

Genera nel programma un salto all'istruzione registrata da PUN, l'esecuzione continuerà a partire da quest'ultima.

Si utilizza per ripetere un numero indeterminato di volte un insieme di istruzioni che sono comprese fra le istruzioni PUN e SAL. Il formato dell'istruzione è SAL xx, essendo xx il numero di volte che bisogna ripetere

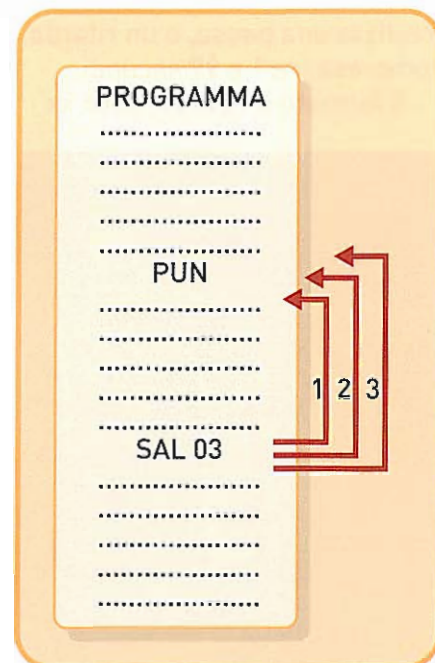


Quando l'operando di SAL è 01 il ciclo fra le istruzioni PUN e SAL 01 si ripete in modo infinito.

le istruzioni posizionate fra PUN e SAL xx. Nel caso in cui l'operando di SAL sia 01 il ciclo si ripete in modo infinito, come si vede nella figura. Se l'operando di SAL è diverso da 01 il ciclo si ripete tante volte quante ne esprime il suo valore, dopo di che continua il suo corso come si può vedere nella figura.

Esempio

Redigeremo un programma nel linguaggio MIKAN I per il robot Armdroid, che inizi chiudendo la pinza, poi ruoti sulla base di 780 passi in senso antiorario, in seguito ruoti per la stessa grandezza in senso orario, ripetendo cinque volte questi ultimi due movimenti. Infine dovrà riaprire la pinza.



Quando l'operando di SAL è diverso da 01 il ciclo si ripete un numero di volte uguale al valore dell'operando.

```
CIE
PUN
MOV -0780, 0, 0, 0
MOV 0780, 0, 0, 0
SAL 05
ABR
FIN
```

Esempio

In questo caso il programma avrà inizio muovendo simultaneamente la base di 300 passi in verso antiorario e il gomito di 100 passi verso l'alto, per terminare aprendo e poi chiudendo la pinza, e ripetendo questa apertura e chiusura in modo continuativo.

```
MOV -0300, 0, 0100, 0
PUN
```

Programmazione di robot

ABR
CIE
SAL 01
FIN

è un numero di due cifre il cui valore è compreso fra 01 e 99.

TEM (TEMPORIZZAZIONE)

Realizza una pausa, o un ritardo, compresa fra 1 e 99 secondi.

Il formato è TEM xx, dove xx

Esempio

In questo caso il programma apre la pinza del robot, dopodiché gira la base di 780 passi in verso antiorario,

fa una pausa di 6 secondi, ripete al contrario la rotazione della base, attende nuovamente 10 secondi, e ripete questi due movimenti con le loro pause corrispondenti per 20 volte e termina il programma chiudendo la pinza.

```
ABR
PUN
MOV 0780, 0, 0, 0
TEM 06
MOV -0780, 0, 0, 0
TEM 10
SAL 20
CIE
FIN
```



Per la scrittura rapida di programmi testuali sono stati progettati dei controller con tastiere adatte alle istruzioni del linguaggio.



Per calcolare i passi corrispondenti alla rotazione di un'articolazione, si utilizza un joystick che produce la rotazione desiderata, mentre il controller visualizza i passi corrispondenti.

FUNZIONE F1

Si utilizza per caricare e scaricare programmi di lavoro su supporti di memorizzazione esterna.

FUNZIONE F2

Permette all'utente di accedere a un controllo manuale dei movimenti del robot. Nelle figure a lato si riporta il pannello frontale del controller di Armdroid in cui si può vedere una tastiera specifica adattata alle istruzioni del MIKAN I, un display di visualizzazione e un joystick per controllare manualmente le rotazioni delle articolazioni, e verificare così i passi corrispondenti. L'esecuzione di F2 restituisce cinque valori che sono: xxxx, wwwv, yyyy, zzzz e vvvv, corrispondenti al numero dei passi delle articolazioni della base, della spalla, del gomito, PITCH e ROLL, rispettivamente.

RI 039

ROBOTICA INDUSTRIALE