

IOAN D. DOROFTEI

ROBOTICĂ
ÎNDRUMAR LUCRĂRI PRACTICE
(revizuit 2016)

LUCRAREA 1

ARHITECTURA UNUI ROBOT ȘI ANALIZA STRUCTURALĂ A SISTEMULUI MECANIC

1.1 Scopul lucrării

Scopul acestei lucrări este de a prezenta studenților componentele de bază ale unui robot, precum și de a efectua analiza structurală a sistemului mecanic, folosind roboții din dotarea laboratorului.

1.2 Arhitectura unui robot de tip manipulator

Componentele de bază ale unui sistem robotizat (Fig. 1.1) pot fi grupate în patru sisteme distincte, fiecare având importanța sa și meritând atenția cuvenită. Aceste sisteme sunt:

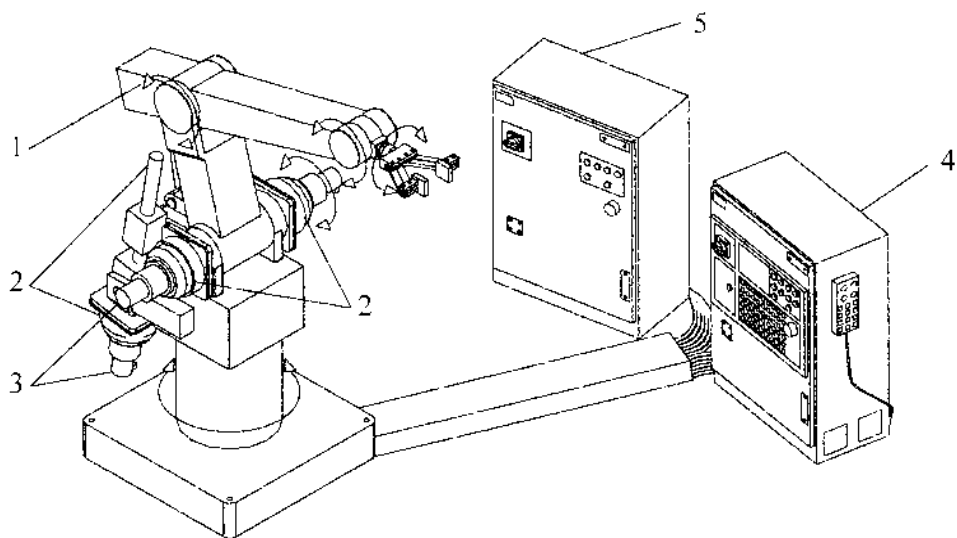


Fig. 1.1 Arhitectura unui robot industrial

- **Sistemul mecanic**, notat cu 1, ce conține lanțuri cinematice (închise sau deschise) - în cazul unui robot de tip manipulator -

sau un corp și roți (picioare) - în cazul unui robot mobil (pășitor). În cazul roboților cu structură mixtă, acest sistem conține un lanț cinematic (închis sau deschis) montat pe o platformă mobilă. Elementele lanțurilor cinematice sunt conectate prin cuple, care determină gradul de mobilitate al robotului. Cuplele cinematice sunt, de regulă, de rotație sau translație.

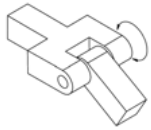


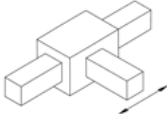


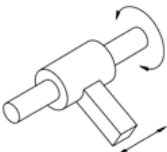


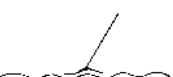
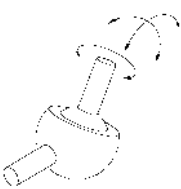
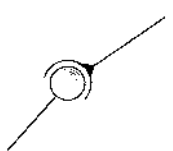
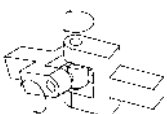
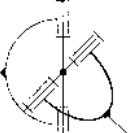
- **Sistemul de acționare** (format din: sursă de energie și amplificator, notate cu 5; motoare și transmisii mecanice, notate cu 2) contribuie la punerea în mișcare a robotului. În mod curent, se utilizează actuatori electrici, hidraulici sau pneumatici, dar pot fi utilizați și actuatori neconvenționali (materiale cu memoria formei, fluide electro-reologice, motoare piezoelectrice polimeri electroactivi, etc). Transmisiiile mecanice fac legătura dintre actuator (motor) și elementele cinematice conducătoare ale sistemului mecanic. Aceste transmisii sunt, în general, utilizate din trei motive: cel mai adesea deoarece caracteristicile actuatorului (turație, cuplu, etc.) nu sunt corespunzătoare pentru acționarea directă a robotului; al doilea motiv ar fi acela că tipul mișcării la ieșirea din actuator nu corespunde tipului cuplei cinematice conducătoare (pentru transformarea mișcării de rotație în mișcare de translație sau invers); al treilea motiv este acela că actuatorii au adesea o masă mare și nu este practică localizarea lor în imediata vecinătate a cuplei cinematice.
- **Sistemul senzorial** (în cazul de față, traductoarele de poziție și de viteză, 3) măsoară poziția și sensul de deplasare pentru elementele cinematice (robotul în ansamblu). Se pot utiliza: traductoare de poziție și de forță în cuplele cinematice, senzori tactili pe degetele de apucare, camere video pe o platformă mobilă, etc.
- **Sistemul de comandă** (control), 4, citește informațiile de la senzori și transmite semnale de control către actuatori, pentru realizarea obiectivelor specificate în program.

1.3 Analiza structurală a sistemului mecanic

1.3.1 Definiții

Elementul cinematic este un corp solid, format dintr-o singură piesă mecanică sau din mai multe piese îmbinate rigid între ele. Elementele cinematice pot fi rigide sau flexibile (cabluri, curele). În lucrarea de față, elementele cinematice vor fi considerate corpuri rigide.

Tabelul 1.1 Principalele tipuri de cuple cinematice

Tipul cuplei cinematice	Forma fizică	Simbol	Grade de mobilitate
Rotație (R)		-plană -spațială  	1
Translație (T)		-plană -spațială  	1
Cilindrică (C)			2
Elicoidală (E)			1
Sferică (S)			3
Universală (U)			2

Legătura directă și mobilă dintre două elemente cinematice poartă numele de *cuplă cinematică*. Numărul de parametri independenți care determină complet poziția relativă a celor două elemente reprezintă *gradul de mobilitate al cuplei, m*, acesta fiind egal

și cu numărul mișcărilor elementare, pe care le poate executa un element cinematic în raport cu celălalt. Principalele tipuri de cuple cinematice întâlnite în structura roboților sunt prezentate în Tabelul 1.1.

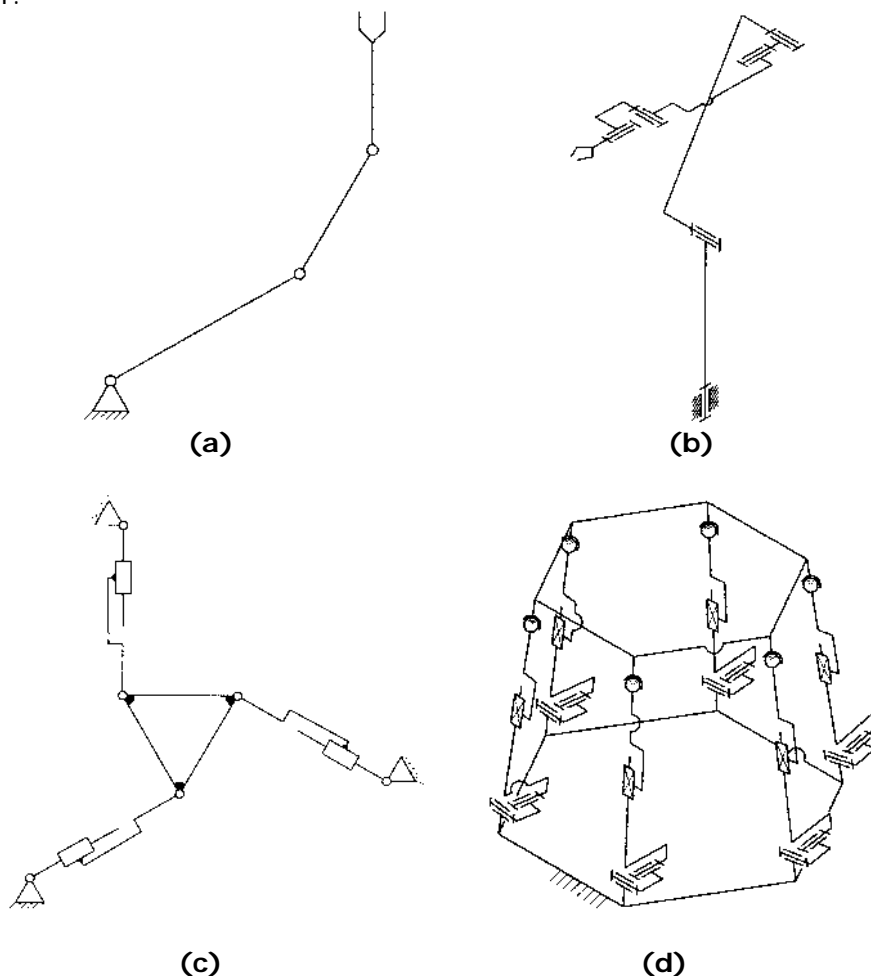


Fig. 1.2 Tipuri de lanțuri cinematice: a) deschis, serial, plan;
b) deschis, serial, spațial; c) închis, paralel, plan;
d) închis, paralel, spațial

Lanțul cinematic este o înșiruire de elemente cinematice, legate între ele prin cuple. *Mecanismul* este un lanț cinematic, care are următoarele proprietăți:

- Conține un element de referință, considerat fix, numit bază sau batiu, în raport cu care are loc mișcarea celorlalte elemente;

- Are un număr de cuple cinematice conducătoare, stabilit astfel încât mișcarea tuturor elementelor să fie determinată.

Mecanismul poate fi constituit din lanțuri cinematice: deschise (Fig. 1.2.a și 1.2.b) sau închise (Fig. 1.2.c și 1.2.d), seriale (Fig. 1.2.a și 1.2.b) sau paralele (Fig. 1.2.c și 1.2.d), plane (Fig. 1.2.a și 1.2.c) sau spațiale (Fig. 1.2.b și 1.2.d).

Un lanț cinematic este denumit *închis* dacă acesta formează o buclă închisă, în caz contrar acesta este numit *deschis*. Dacă fiecare element al lanțului cinematic, exceptând primul și ultimul, este legat la alte două elemente cinematice, acesta se numește lanț cinematic *serial* (*deschis*). Un lanț cinematic este *paralel* dacă primul și ultimul element sunt legate între ele prin intermediul a cel puțin două lanțuri cinematice independente. Un lanț cinematic ale cărui elemente efectuează mișcări în același plan sau în planuri paralele este numit lanț cinematic *plan*. Dacă mișcările elementelor au loc în cel puțin două planuri neparalele, acesta se va numi lanț cinematic *spațial*.

1.3.2 Structura sistemului mecanic

Sistemul mecanic al unui robot de tip manipulator are în structura sa o serie de mecanisme și dispozitive, care asigură apucarea obiectului manipulat, respectiv poziționarea și orientarea în spațiu a acestuia (sau a unei scule) - Fig. 1.3.

Mecanismul de ghidare are rolul de a conferi efectorului final mișcările necesare situării (poziționării și orientării) în spațiu a obiectului manipulat, precum și energia mecanică necesară efectuării acestor mișcări. Poziționarea (deplasarea) obiectului manipulat o realizează *mecanismul de poziționare (generator de traiectorie)*, iar orientarea acestui obiect este asigurată de *mecanismul de orientare* (acesta conține numai cuple cinematice de rotație). De multe ori, mecanismul de poziționare este numit braț, iar mecanismul de orientare este numit articulație carpiană sau mecanism carpian.

Efectorul final poate fi un *dispozitiv de prehensiune* - cu rolul de a fixa obiectul manipulat de ultimul element al mecanismului de ghidare - sau o *sculă* (spre exemplu, un bisturiu).

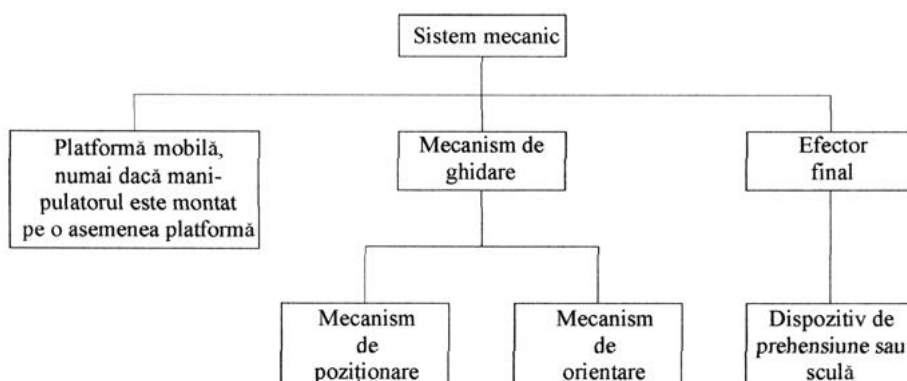


Fig. 1.3 Structura sistemului mecanic

1.3.3 Mobilitatea sistemului mecanic

Gradul de mobilitate al unui mecanism este dat de numărul minim de parametri independenți necesari pentru a specifica poziția tuturor elementelor mobile ale acestuia. De asemenea, gradul de mobilitate al mecanismului este egal și cu numărul cuplelor cinematice conducătoare din structura acestuia.

Presupunem că sistemul mecanic al unui robot este constituit dintr-un lanț cinematic având n elemente. În cazul în care toate aceste elemente cinematice sunt libere (să se miște independent în spațiu), gradul de mobilitate al sistemului va fi $6n$ (știind că un corp liber în spațiul tridimensional poate avea trei rotații și trei translații). Dacă unul dintre aceste elemente este fixat, pentru ca lanțul cinematic să fie mecanism, acest element își va pierde gradele sale de libertate. De aceea, gradul de mobilitate al sistemului va deveni $6 \cdot (n - 1)$, atâta vreme cât elementele acestuia nu sunt legate între ele prin cuple cinematice. Dacă două elemente cinematice ale sistemului formează o cuplă cu gradul de mobilitate m_i , gradul de mobilitate al sistemului se diminuează, datorită faptului că cele două elemente (care aveau fiecare câte 6 grade de libertate) au mai pierdut din numărul gradelor de libertate, prin conectarea lor. Aceasta înseamnă că gradul de mobilitate al sistemului se reduce cu $6 - m_i$ (cele două elemente cinematice au acum, împreună, 6 grade de libertate în spațiu, la care se adaugă mișcările relative m_i dintre acestea, permise de cupla; inițial ele aveau împreună 12 grade de libertate). Dacă se va continua

Robotică-Îndrumar de lucrări practice

cu legarea elementelor până se obțin k cuple cinematice, gradele de mobilitate pierdute de sistem pot fi calculate astfel:

$$(6 - m_1) + (6 - m_2) + \dots + (6 - m_k) = \sum_{i=1}^k (6 - m_i) = 6 \cdot k - \sum_{i=1}^k m_i$$

În aceste condiții gradul de mobilitate al mecanismului va fi dat de relația:

$$M = 6 \cdot (n - 1) - \left(6 \cdot k - \sum_{i=1}^k m_i \right) = 6 \cdot (n - k - 1) + \sum_{i=1}^k m_i \quad (1.1)$$

unde: n – numărul total de elemente cinematice; k – numărul total de cuple cinematice; m_i - gradul de mobilitate al cuplei i .

Această modalitate de calcul a gradului de mobilitate este numită *principiul lui Kutzbach*. Dacă manipulatorul are numai cuple cinematice inferioare, cu gradul de mobilitate $m = 1$ (cuple de rotație, translație sau elicoidale), relația (1.1) devine:

$$M = 6 \cdot (n - k - 1) + k = 6 \cdot (n - 1) - 5k \quad (1.2)$$

Pentru un manipulator serial (deschis), numărul cuplelor cinematice este $k = n - 1$, relația (1.1) scriindu-se sub forma:

$$M = 6 \cdot [n - (n - 1) - 1] + \sum_{i=1}^k m_i = \sum_{i=1}^k m_i \quad (1.3)$$

Dacă manipulatorul serial are numai cuple cinematice inferioare, cu gradul de mobilitate $m = 1$, relația (1.3) devine:

$$M = k = n - 1 \quad (1.4)$$

Pentru un mecanism plan, aplicând aceeași metodă, se obține următoarea relație pentru calculul gradului de mobilitate:

$$M = 3 \cdot (n - k - 1) + \sum_{i=1}^k m_i \quad (1.5)$$

Dacă manipulatorul plan conține un lanț cinematic serial, atunci $k = n - 1$, relația (1.5) devenind identică cu (1.3). Dacă toate cuplele cinematice sunt inferioare, calculul gradului de mobilitate se va face utilizând relația (1.4).

Pentru a deplasa și orienta un obiect în spațiul tridimensional, mecanismul de ghidare trebuie să aibă cel puțin 6 grade de mobilitate: trei grade de mobilitate asigurate de mecanismul de poziționare, pentru deplasarea obiectului manipulat în lungul celor trei axe de coordonate; trei grade de mobilitate asigurate de mecanismul de orientare, pentru orientarea obiectului manipulat în raport cu cele

Robotică-Îndrumar de lucrări practice

trei axe de coordonate. În cazul în care efectorul final este un mecanism de prehensiune, apare al 7-lea grad de mobilitate, necesar strângerii și fixării obiectului manipulat.

În funcție de sarcina pe care o are de îndeplinit robotul, pot exista și următoarele situații:

- dacă operația de manipulare este mai simplă, gradul de mobilitate poate fi mai mic decât 6, respectiv $M < 6$;

- pentru operații mai complexe, când obiectul manipulat trebuie să evite anumite obstacole, gradul de mobilitate trebuie să fie mai mare decât 6, $M > 6$.

Diferența $M - 6$ ne dă gradul de manevrabilitate al robotului:

$$M_m = M - 6 \quad (1.6)$$

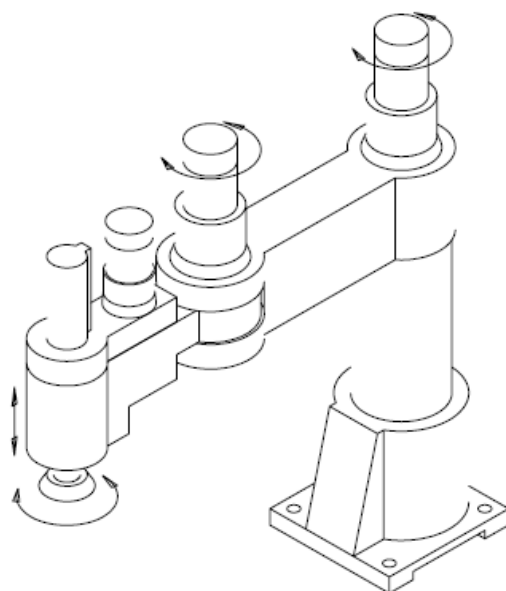
Roboții ale căror mecanisme de ghidare se caracterizează prin $M_m < 0$ se numesc *roboți deficienți*, iar cei ale căror mecanisme de ghidare au $M_m > 0$ se numesc *roboți redundanți*.

1.3.4 Exemplu de calcul

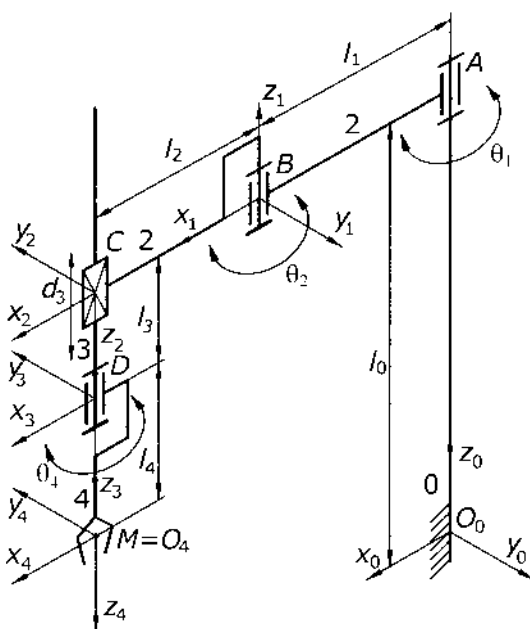
Pentru a exemplifica modul în care trebuie efectuată analiza structurală a sistemului mecanic, vom considera robotul de tip manipulator din Fig. 1.4.a, robot cu structură tip SCARA (*Selected Compliance Articulated Robot Arm*).

Se analizează foarte atent sistemul mecanic al robotului, astfel încât să se determine tipul și numărul cuplelor cinematice din structura acestuia. Transmisiile mecanice auxiliare, utilizate pentru transmiterea mișcării de la motoare la elementele de bază ale sistemului mecanic, respectiv dispozitivul de prehensiune, nu se iau în considerare.

Se reprezintă schema cinematică a robotului la scară (măsurând lungimile elementelor cinematice), pentru exemplul nostru aceasta fiind prezentată în Fig. 1.4.b. Se notează elementele robotului cu cifre și cuplele cinematice cu litere mari, plecând de la elementul fix către efectorul final. Dimensiunile elementelor sunt trecute într-un tabel (Tabelul 2.1).



(a)



(b)

Fig. 1.4 Robot tip SCARA: a) model CAD;
b) schemă cinematică

Tabelul 2.1

Nr.crt.	Notatia dimensiunii masurate	Dimensiunea reala [m]	Dimensiunea reprezentativa [mm]
1	l_0	0.6	60
2	l_1	0.4	40
3	l_2	0.3	30
4	l_3	0	0
5	l_4	0.15	15

Pentru exemplul considerat, numărul total al elementelor cinematice este $n = 5$, iar numărul total al cuplelor cinematice pentru mecanismul de ghidare este $k = 4$ (trei cuple cinematice – A , B , C – în structura mecanismului de poziționare și o cuplă – D – în structura mecanismului de orientare. Toate cuplele cinematice sunt inferioare (A , B și D - rotație, C - translație), având gradul de mobilitate $m = 1$.

Folosind relația (1.1), vom calcula gradul de mobilitate M al robotului:

$$M = 6 \cdot (5 - 4 - 1) + \sum_{i=1}^4 m_i = 0 + 4 = 4$$

Acest rezultat conduce la concluzia că numărul parametrilor cinematici independenți necesari pentru a determina mișcarea celor patru elemente mobile este 4 (numărul motoarelor necesare pentru acționarea robotului va fi 4).

1.4 Modul de lucru

- Conducătorul de lucrări va prezenta sistemele componente ale unui robot, utilizând echipamentele din dotarea laboratorului;
- Studenții vor reprezenta la scară schema cinematică, trecând dimensiunile în Tabelul 2.1, și vor efectua analiza structurală a sistemului mecanic pentru un robot, specificat de conducătorul de lucrări (un robot din dotarea laboratorului).
- Schema cinematică va fi folosită la lucrările următoare, pentru analiza cinematică și dinamică a robotului respectiv.

LUCRAREA 2

TRANSFORMĂRI DE COORDONATE

2.1 Scopul lucrării

Scopul acestei lucrări este de a deprinde studenții cu modul de efectuare a calculelor necesare determinării poziției și orientării unui corp în spațiul tridimensional, în raport cu un sistem de axe de referință.

2.2 Baze teoretice

2.2.1 Sisteme de coordonate

Pozițiile și orientările relative ale elementelor cinematice, ale unui robot de tip manipulator, sunt definite de parametrii variabili ai cuplelor cinematice. Deoarece fiecare element reprezintă un corp rigid, este necesar ca acestuia să i se atașeze un sistem de coordonate cartezian. În aceste condiții, configurația robotului este definită de pozițiile și orientările relative ale sistemelor de axe atașate fiecărui element cinematic.

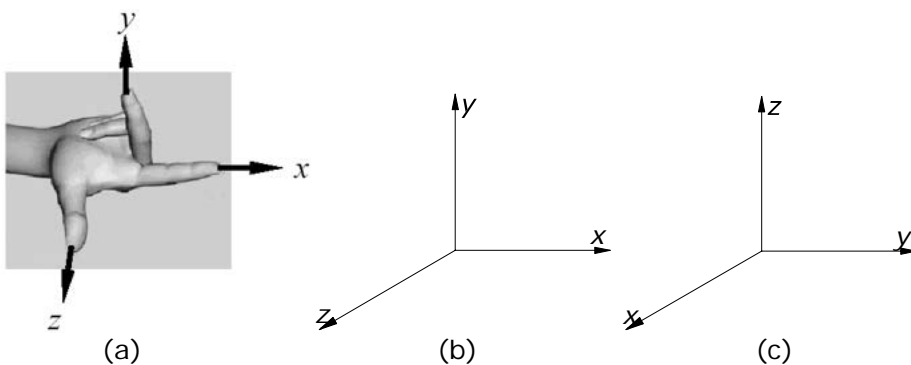


Fig. 2.1 Sisteme de coordonate

În robotică, sistemele de coordonate utilizate respectă regula mâinii drepte (vezi Fig. 2.1). Sensurile pozitive ale axelor unui sistem cartezian ce respectă această regulă se obține astfel: ținând deschisă

mâna dreaptă, cu degetul mare desfăcut, direcția palmei (celor patru degete) va indica sensul pozitiv al axei x , iar degetul mare va indica sensul pozitiv al axei z . Sensul pozitiv al axei y se va obține îndoind cu 90° cele patru degete.

2.2.2 Poziția și orientarea unui corp rigid

Considerăm un sistem de referință $x_{n-1}, y_{n-1}, z_{n-1}$ și un corp rigid, căruia i se atașează sistemul de coordonate x_n, y_n, z_n (Fig. 2.2). Coordonatele unui punct oarecare P de pe corp, în raport cu originea O_{n-1} a sistemului de referință, pot fi calculate cu relațiile:

$$\begin{cases} x_{n-1} = x + x_n \cdot \cos(x_{n-1}, x_n) + y_n \cdot \cos(x_{n-1}, y_n) + z_n \cdot \cos(x_{n-1}, z_n) \\ y_{n-1} = y + x_n \cdot \cos(y_{n-1}, x_n) + y_n \cdot \cos(y_{n-1}, y_n) + z_n \cdot \cos(y_{n-1}, z_n) \\ z_{n-1} = z + x_n \cdot \cos(z_{n-1}, x_n) + y_n \cdot \cos(z_{n-1}, y_n) + z_n \cdot \cos(z_{n-1}, z_n) \end{cases} \quad (2.1)$$

unde: $x_{n-1}, y_{n-1}, z_{n-1}$ reprezintă coordonatele punctului P în raport cu originea O_{n-1} a sistemului de referință; x_n, y_n, z_n sunt coordonatele aceluiasi punct în raport cu originea O_n a sistemului de axe atașat corpului; x, y, z reprezintă coordonatele punctului O_n în raport cu originea O_{n-1} a sistemului de referință.

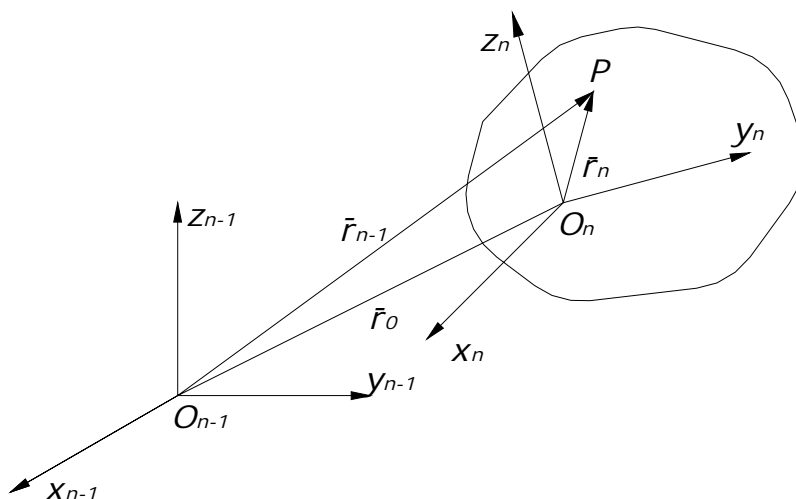


Fig. 2.2 Transformare de coordonate

Sub formă matriceală, relațiile (2.1) pot fi scrise astfel:

$${}^{n-1}P = {}^{n-1}_nTr + {}^{n-1}_nR \cdot {}^nP \quad (2.2)$$

unde:

$${}^{n-1}P = \begin{bmatrix} x_{n-1} \\ y_{n-1} \\ z_{n-1} \end{bmatrix}; \quad {}^{n-1}_nTr = \begin{bmatrix} x \\ y \\ z \end{bmatrix}; \quad {}^nP = \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix};$$

$${}^{n-1}_nR = \begin{bmatrix} \cos(x_{n-1}, x_n) & \cos(x_{n-1}, y_n) & \cos(x_{n-1}, z_n) \\ \cos(y_{n-1}, x_n) & \cos(y_{n-1}, y_n) & \cos(y_{n-1}, z_n) \\ \cos(z_{n-1}, x_n) & \cos(z_{n-1}, y_n) & \cos(z_{n-1}, z_n) \end{bmatrix} \quad (2.3)$$

Poziția corpului rigid, căruia i s-a atașat sistemul de axe x_n, y_n, z_n , în raport cu sistemul de referință $x_{n-1}, y_{n-1}, z_{n-1}$ este dată de matricea ${}^{n-1}_nTr$, iar orientarea acestuia este definită de matricea ${}^{n-1}_nR$.

2.2.3 Matricea de rotație

Matricea ${}^{n-1}_nR$, ce definește orientarea unui sistem de axe (a unui corp rigid) în raport cu un sistem de axe de referință, poartă numele de matrice de rotație.

2.2.3.1 Rotații elementare (simple)

Considerăm cazul particular al rotației sistemului de coordonate $x_{n-1}, y_{n-1}, z_{n-1}$ în jurul axei x_{n-1} , cu unghiul θ_x (Fig. 2.3).

Matricea de transformare corespunzătoare rotației menționate se va obține particularizând matricea ${}^{n-1}_nR$, relația (2.3), la cazul nostru.

$$R_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x & \cos \theta_x \end{bmatrix} \quad (2.4)$$

Similar, pentru rotația în jurul axelor y_{n-1}, z_{n-1} , cu unghiurile θ_y, θ_z , matricele elementare de rotație sunt:

$$\mathbf{R}_y(\theta_y) = \begin{bmatrix} \cos \theta_x & 0 & \sin \theta_x \\ 0 & 1 & 0 \\ -\sin \theta_x & 0 & \cos \theta_x \end{bmatrix} \quad (2.5)$$

$$\mathbf{R}_z(\theta_z) = \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

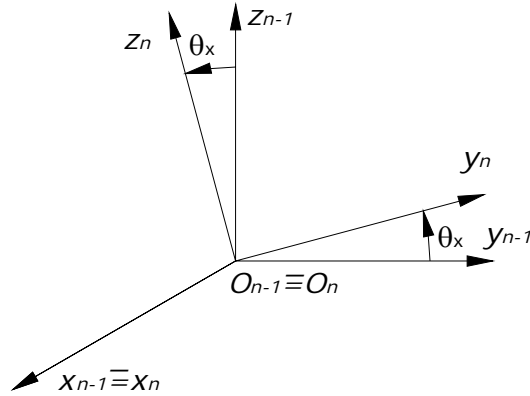


Fig. 2.3 Rotație în jurul axei x

Aceste matrice sunt utile pentru descrierea rotațiilor corpurilor rigide în jurul unei axe arbitrare în spațiul tridimensional. Ele au fost determinate considerând că rotațiile au fost făcute în sens pozitiv (spre dreapta).

Este ușor de verificat faptul că aceste matrice verifică relația:

$$\mathbf{R}(-\theta) = \mathbf{R}^T(\theta) \quad (2.7)$$

unde $\mathbf{R}^T(\theta)$ este transpusa matricei de rotație (pentru rotația simplă cu unghiul θ).

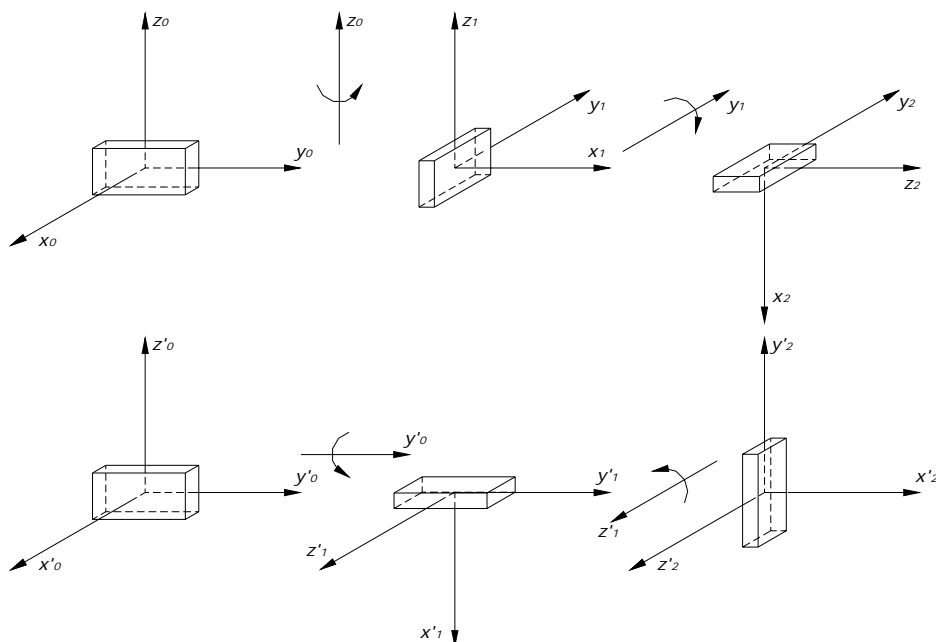
Cu alte cuvinte, modificarea sensului de rotație conduce la altă matrice de transformare, lucru ce trebuie reținut și de care trebuie să se țină seama.

2.2.3.2 Compunerea matricelor de rotație

Rotația totală a unui corp rigid în spațiul tridimensional poate fi exprimată ca o succesiune de rotații parțiale. Fiecare rotație este

definită în raport cu cea precedentă, iar sistemul de axe la care se raportează rotația parțială poartă numele de sistem de axe curent.

Considerăm pentru exemplificare un corp rigid ce suferă două rotații succesive (a câte 90° fiecare) în jurul a două axe ale sistemului curent (Fig. 2.4).



**Fig. 2.4 Rotații succesive: a) succesiunea $Z_0 \rightarrow Y_1$;
 b) succesiunea $Y'_0 \rightarrow Z'_1$**

Pentru succesiunea de rotații prezentată în Fig. 2.4.a, matricele elementare de rotație sunt:

$${}^0_1R = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad {}^1_2R = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

iar matricea totală de rotație va fi:

$${}^0_2R = {}^0_1R \cdot {}^1_2R = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

Pentru succesiunea de rotații din Fig. 2.4.b, matricele elementare de rotație sunt:

$${}^{0'}_1R = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad {}^{1'}_2R = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

iar matricea totală de rotație corespunzătoare:

$${}^{0'}_2R = {}^{0'}_1R \cdot {}^{1'}_2R = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.11)$$

Se poate constata cu ușurință că rotațiile parțiale succesive nu sunt comutative, fapt foarte important și căruia trebuie să i se acorde atenția cuvenită.

2.2.4 Matricea omogenă de transformare

Considerăm din nou corpul rigid din Fig. 2.3, a cărui poziție și orientare sunt raportate la sistemul de referință $x_{n-1} y_{n-1} z_{n-1}$.

Transformarea de coordonate scrisă sub forma (2.2) prezintă unele dezavantaje din punct de vedere al analizei cinemateice, datorită caracterului neomogen (conține termenul liber ${}^{n-1}_nTr$).

De asemenea, o matrice de ordinul 3x3 (cum este matricea ${}^{n-1}_nR$) poate fi utilizată pentru descrierea rotației corpului rigid, dar nu și a translației acestuia, termenul ${}^{n-1}_nTr$ nefiind inclus în această matrice. Ori, în robotică, descrierea translațiilor (poziționării obiectului manipulat) este foarte importantă. Translația poate fi descrisă prin introducerea unei a patra coloane în matricea ${}^{n-1}_nR$, însă matricea rezultantă nu este pătrată și nu se poate scrie inversa ei.

Robotică - Îndrumar de lucrări practice

Pentru a elimina aceste dezavantaje, respectiv pentru omogenizarea matricei de transformare, la cele trei ecuații (2.1) se adaugă ecuația formală

$$t_{n-1} = t_n = 1 \quad (2.12)$$

În aceste condiții, transformarea de coordonate sub formă matriceală se scrie astfel:

$${}^{n-1}P = {}^{n-1}T_n \cdot {}^nP \quad (2.13)$$

unde:

$${}^{n-1}P = \begin{bmatrix} x_{n-1} \\ y_{n-1} \\ z_{n-1} \end{bmatrix}; \quad {}^nP = \begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix}; \quad (2.14)$$

$${}^{n-1}T_n = \begin{bmatrix} \cos(x_{n-1}, x_n) & \cos(x_{n-1}, y_n) & \cos(x_{n-1}, z_n) & x \\ \cos(y_{n-1}, x_n) & \cos(y_{n-1}, y_n) & \cos(y_{n-1}, z_n) & y \\ \cos(z_{n-1}, x_n) & \cos(z_{n-1}, y_n) & \cos(z_{n-1}, z_n) & z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^{n-1}R_n & \vdots & {}^{n-1}Tr_n \\ \cdots & \cdots & \cdots \\ 0 & \vdots & 1 \end{bmatrix}$$

2.2.4.1 Forme particulare pentru matricea de transformare omogenă

În cazul în care sistemul de axe $x_{n-1} y_{n-1} z_{n-1}$ comportă doar mișcări de rotație în jurul axelor acestuia, matricea de transformare omogenă capătă următoarea formă particulară:

$$\begin{aligned} {}^{n-1}T_n &= \begin{bmatrix} \cos(x_{n-1}, x_n) & \cos(x_{n-1}, y_n) & \cos(x_{n-1}, z_n) & 0 \\ \cos(y_{n-1}, x_n) & \cos(y_{n-1}, y_n) & \cos(y_{n-1}, z_n) & 0 \\ \cos(z_{n-1}, x_n) & \cos(z_{n-1}, y_n) & \cos(z_{n-1}, z_n) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\ &= \begin{bmatrix} {}^{n-1}R_n & \vdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \vdots & 1 \end{bmatrix} \end{aligned} \quad (2.15)$$

Pentru o rotație a acestui sistem în jurul axei x_{n-1} , cu unghiul θ_x , matricea de transformare corespunzătoare rotației menționate se va obține particularizând matricea ${}^{n-1}T_n$ la cazul nostru.

$$\mathbf{R}_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x & 0 \\ 0 & \sin \theta_x & \cos \theta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.16)$$

Similar, pentru rotația în jurul axelor y_{n-1} , z_{n-1} , cu unghiurile θ_y , θ_z , matricea capătă formele:

$$\mathbf{R}_y(\theta_y) = \begin{bmatrix} \cos \theta_y & 0 & \sin \theta_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.17)$$

$$\mathbf{R}_z(\theta_z) = \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 & 0 \\ \sin \theta_z & \cos \theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

Dacă presupunem că sistemul de axe $x_n y_n z_n$ comportă doar translații în raport cu sistemul de referință $x_{n-1} y_{n-1} z_{n-1}$, în absența rotațiilor matricea de transformare omogenă (2.14) capătă forma:

$${}^{n-1}_n\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

Considerând cazul particular al translației sistemului de axe $x_n y_n z_n$ în lungul axei x_{n-1} , cu mărimea x (Fig. 2.5), matricea de translație corespunzătoare va fi:

$$\mathbf{Tr}_x = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.20)$$

Similar, translațiile elementare în lungul axelor y_{n-1} , z_{n-1} , cu mărimile y , z , sunt exprimate cu ajutorul matricelor:

$$\mathbf{Tr}_y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad \mathbf{Tr}_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.21)$$

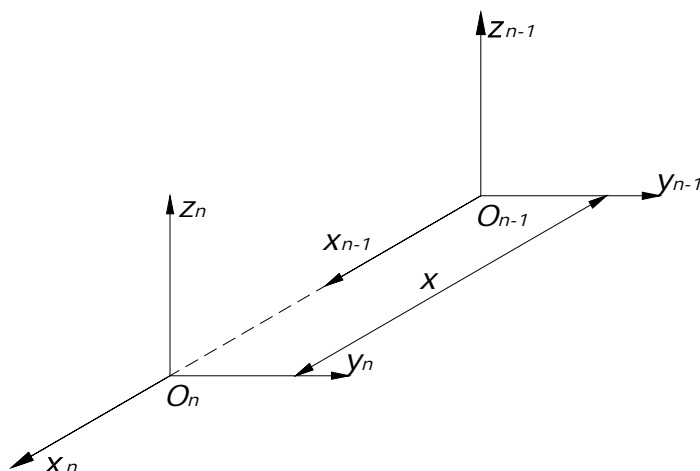


Fig. 2.5 Translație în jurul axei x

2.3 Exemplu

2.3.1 Introducere în MATLAB

Deoarece efectuarea calculelor privind transformările de coordonate, dar și a calculelor aferente altor lucrări practice, se va face în MATLAB, în cele ce urmează vor fi prezentate câteva elemente ajutătoare, în sprijinul studenților ce nu au experiență în utilizarea acestui soft.

Ținând seama de faptul că în majoritatea cazurilor vom avea de a face cu calcule matriceale, este util să se cunoască modul de scriere a unei matrice în acest limbaj și modul de efectuare a principalelor operații cu matrice.

```
A=[1 3 0]; % Matrice cu o linie
B=[2;4;1];
sau
B=[2
```

Robotică - Îndrumar de lucrări practice

```
4
1];
sau
B=[2 4 1]';           % Matrice cu o coloană

C=[2 0 1
0 3 0]
1 2 0];
sau
C=[2 0 1;0 3 0;1 2 0]; % Matrice cu 3 linii și 3 coloane (3x3)

D=[0 1 4
1 1 0
2 1 0]
E=C+D;                % Suma a două matrice (C+D=D+C)

F=C-D;                % Diferența a două matrice [C-D=-(D-
C)]

G=C*D;                % Produsul a două matrice (C*D≠D*C)

F=C/D;                % Raportul a două matrice (C/D≠D/C)

dC=det(C);            % Determinantul matricei C

invC=inv(C);           % Inversa matricei C

CT=C';                % Transpusa matricei C
```

2.3.2 Exemplu de calcul

Considerăm punctul P , aparținând unui corp rigid, de coordonate $(2, 4, 1)$ în raport originea sistemului de axe atașat acestuia. Corpul respectiv comportă mai întâi o rotație cu 30° în jurul axei y_0 , apoi o translație cu 50 mm în jurul axei x_1 , respectiv o rotație cu 60° în jurul axei z_2 (Fig. 2.6). Se cer coordonatele finale ale punctului P în raport cu sistemul de referință x_0, y_0, z_0 .

Matricele omogene elementare, la trecerea de la sistemul de referință x_0, y_0, z_0 la sistemul de coordonate x_3, y_3, z_3 , sunt următoarele:

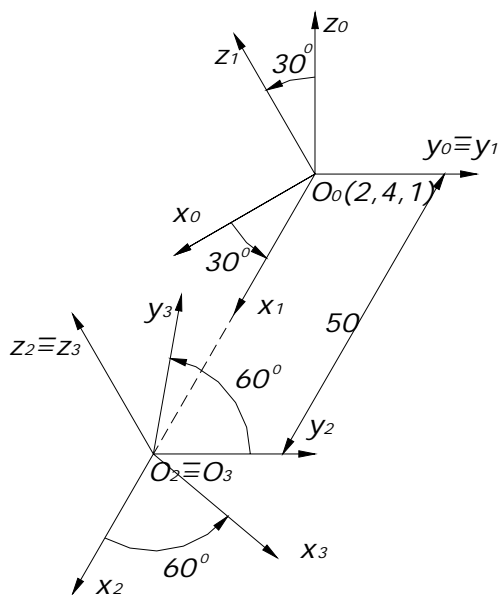


Fig. 2.6 Exemplu

$${}^0_1T = R_y(30^\circ) = \begin{bmatrix} \frac{\sqrt{3}}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad {}^1_2T = Tr_x = \begin{bmatrix} 1 & 0 & 0 & 50 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

$${}^2_3T = R_z(60^\circ) = \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 & 0 \\ \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matricea omogenă totală va fi:

$${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T = \begin{bmatrix} 0.433 & -0.75 & 0.5 & 43.3013 \\ 0.866 & 0.5 & 0 & 0 \\ -0.25 & 0.433 & 0 & -25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Coordonatele punctului P în raport cu sistemul de axe x_3, y_3, z_3 se știu ca fiind ${}^3P = \begin{bmatrix} 2 \\ 4 \\ 1 \\ 1 \end{bmatrix}$. Matricea ce conține coordonatele

punctului respectiv în raport cu sistemul de referință vor fi:

$${}^0P = {}^0_3T \cdot {}^3P = \begin{bmatrix} 41.6673 \\ 3.7321 \\ -22.9019 \\ 1 \end{bmatrix}$$

Din matricea 0P rezultă coordonatele cerute ca fiind (41.6673, 3.7321, -22.9019).

2.3.3 Program de calcul în MATLAB

În cele ce urmează, va fi prezentat un exemplu de program pentru efectuarea calculelor făcute în paragraful precedent.

```
clear all
```

```
P3 = [2
      4
      1
      1]
```

```
ty=30*pi/180
Ry = [cos(ty)  0  sin(ty)  0
      0        1  0        0
      -sin(ty) 0  cos(ty)  0
      0        0  0        1]
```

```
x=50
```

```
Tx = [1  0  0  x
       0  1  0  0
       0  0  1  0
       0  0  0  1]
```

```
tz=60*pi/180
```

```
Rz = [cos(tz)  -sin(tz)  0  0
       sin(tz)   cos(tz)  0  0
       0          0       0  0
       0          0       0  1]
```

```
T = Ry * Tx * Rz
```

```
P0 = T * P3
```

2.4 Modul de lucru

- Conducătorul de lucrări va da studenților coordonatele punctului P aparținând unui corp rigid, coordonate raportate la sistemul de coordonate final x_n, y_n, z_n , precum și cele n succesiuni de transformări elementare suferite de corpul rigid, plecând de la un sistem de referință x_0, y_0, z_0 ;

- Studenții vor scrie un program în MATLAB, pentru a determina coordonatele punctului P în raport cu sistemul de referință x_0, y_0, z_0 . Pentru aceasta, ei se vor folosi de exemplul prezentat în paragraful anterior.

LUCRAREA 3

ANALIZA CINEMATICALĂ DIRECTĂ A POZIȚIILOR UNUI ROBOT SPAȚIAL DE TIP MANIPULATOR

3.1 Scopul lucrării

Scopul acestei lucrări este de a efectua analiza cinematică directă a pozițiilor pentru un robot spațial de tip manipulator, folosind *parametrii Denavit-Hartenberg* (D-H) *standard*. Astfel, vor fi determinați parametrii D-H ai manipulatorului și se va efectua doar un calcul simbolic, cu scopul de a deprinde studenții cu determinarea parametrilor respectivi și cu pașii ce trebuie urmați, în vederea efectuării analizei cinemateice directe a pozițiilor.

3.2 Baze teoretice

3.2.1 Generalități

Analiza cinematică directă a pozițiilor unui robot de tip manipulator necesită cunoașterea parametrilor cinemateici ai fiecărei cuple. Cunoscând acești parametri, se pot determina poziția și orientarea efectorului final, respectiv poziția și orientarea obiectului manipulat. Pentru a putea efectua această analiză, se atașează câte un sistem de axe cartezian fiecărui element cinematic și se determină parametrii *Denavit - Hartenberg* (D-H) corespunzători.

În literatura de specialitate există două convenții privind determinarea parametrilor D-H:

- Parametrii D-H standard, prezentați de Paul [1];
- Parametrii D-H utilizați de Craig (parametrii D-H modificați) [2].

Ambele convenții de stabilire a parametrilor D-H sunt valabile, alegerea uneia dintre acestea fiind o problemă de obișnuință sau preferință.

În continuare, se va prezenta descrierea cinematică utilizând parametrii D-H după convenția Paul. Modelarea cinematică se face în ipoteza că precizia geometrică a robotului este absolută.

3.2.2 Definirea parametrilor Denavit-Hartenberg standard

Atașarea sistemelor de axe se poate face arbitrar, dar este mai convenabil să se respecte anumite reguli, în cazul nostru regulile impuse de convenția Denavit-Hartenberg.

Considerăm două elemente $i-1$ și i , legate prin cupla cinematică i (Fig. 3.1). Pentru definirea sistemului de axe atașat elementului i , va fi utilizată convenția menționată, după cum urmează:

- Axa z_i se alege astfel încât aceasta să coincidă cu axa cuplei cinematice $i + 1$.
- Originea O_i a sistemului va fi localizată la intersecția axei z_i cu perpendiculara comună la axele z_{i-1} și z_i .
- Axa x_i se alege în lungul perpendicularei comune la axele z_{i-1} și z_i , având sensul orientat de la cupla i spre cupla $i + 1$.
- Axa y_i rezultă astfel încât sistemul de coordonate să respecte regula mâinii drepte.

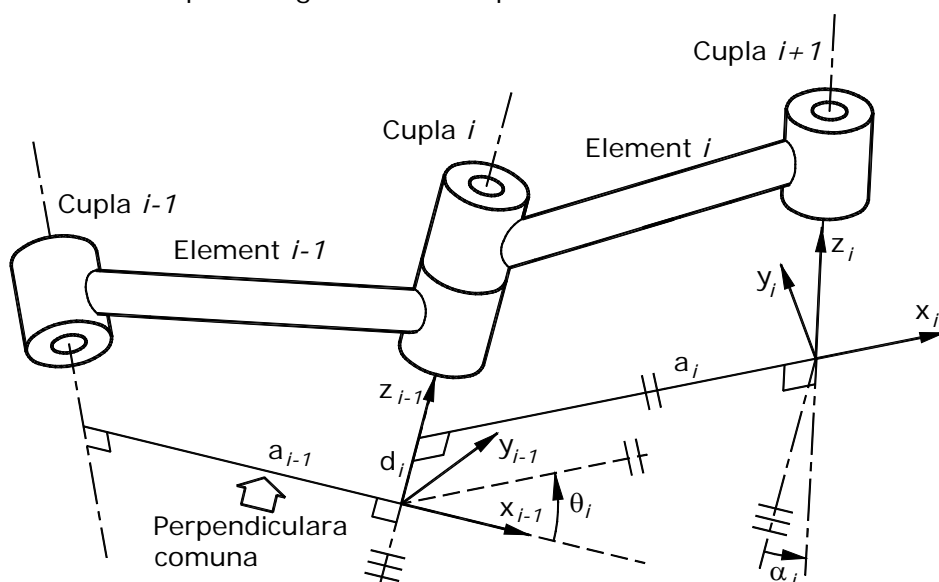


Fig. 3.1 Definirea parametrilor Denavit-Hartenberg standard

Convenția Denavit-Hartenberg standard oferă mai multe soluții în următoarele cazuri:

- Pentru sistemul inițial $\{0\}$, este specificată numai axa z_0 . Originea O_0 și axa x_0 pot fi alese arbitrar. Axa x_0 se alege, totuși, astfel încât matricea de transformare rezultată la trecerea de la sistemul de axe $\{0\}$ la sistemul $\{1\}$ să fie cât mai simplă.
- Pentru sistemul n , deoarece nu există cupla cinematică $n+1$, axa z_n nu este definită unic, atâta timp cât axa x_n trebuie să fie perpendiculară pe axa z_{n-1} . În mod obișnuit, cupla n este de rotație, de aceea axa z_n va fi paralelă cu axa z_{n-1} .
- Când două axe succesive sunt paralele, perpendiculara lor comună nu este unic definită.
- Când două axe succesive se intersectează, direcția axei x_i este arbitrară.
- Când cupla i este de translație, direcția axei z_{i-1} este arbitrară.

În astfel de cazuri, atașarea sistemelor de axe se face în așa fel încât problema să fie cât mai simplă; spre exemplu axele a două sisteme succesive pot fi paralele.

Odată ce au fost stabilite sistemele de coordonate, poziția și orientarea sistemului i în raport cu sistemul $i-1$ sunt definite complet de următorii parametri:

a_i - distanța dintre z_{i-1} și z_i , măsurată în lungul lui x_i (a_i este pozitiv când x_{i-1} este orientat de la axa i la axa $i+1$);

α_i - unghiul dintre axele z_{i-1} și z_i , măsurat în jurul axei x_i ; de regulă, acesta este pozitiv când rotația este făcută în sens trigonometric;

d_i - distanța dintre x_{i-1} și x_i , măsurată în lungul lui z_{i-1} ;

θ_i - unghiul dintre axele x_{i-1} și x_i , măsurat în jurul axei z_{i-1} ; de regulă, acesta este pozitiv când rotația are loc în sens trigonometric.

Pentru determinarea semnelor unghiurilor α_i și θ_i , poate fi considerată pozitivă și rotația în sens orar. Adoptarea sensului de rotație pozitiv rămâne la latitudinea persoanei ce efectuează analiza cinematică. Totuși, sensul adoptat trebuie să fie același pentru

Robotică - Îndrumar de lucrări practice

definirea tuturor parametrilor D-H, în cazul aceleiași structuri cinematice.

Doi dintre acești parametri (d_i și a_i) sunt constanți și depind numai de geometria robotului. Dintre ceilalți doi parametri, doar unul este variabil, în funcție de tipul cuplei cinematice:

- θ_i este variabil în cazul în care cupla i este de rotație;
- d_i este variabil în cazul când cupla i este de translație.

Trecerea de la sistemul de referință $i-1$ la sistemul i are loc prin:

- o translație cu d_i în lungul axei z_{i-1} , și o rotație cu θ_i în jurul axei z_{i-1} , urmate de
- o translație cu a_i în lungul axei x_i și o rotație în jurul lui x_i cu unghiul α_i .

Corespunzător celor două etape de trecere de la sistemul de axe $\{i-1\}$ la sistemul $\{i\}$ se poate scrie matricea omogenă de transformare totală, sub forma:

$${}^{i-1}_i\mathbf{T} = \text{Transl}(z_{i-1}, d_i) \cdot \text{Rot}(z_{i-1}, \theta_i) \cdot \text{Transl}(x_i, a_i) \cdot \text{Rot}(x_i, \alpha_i) \quad (3.1)$$

respectiv

$$\begin{aligned} {}^{i-1}_i\mathbf{T} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \\ &\cdot \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\ &= \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (3.2)$$

Matricea de transformare omogenă a unui robot cu n elemente, scrisă în raport cu originea O_0 a sistemului de referință, va fi:

$${}^0_n\mathbf{T} = {}^0_1\mathbf{T} \cdot {}^1_2\mathbf{T} \cdot \dots \cdot {}^{j-1}_j\mathbf{T} \cdot \dots \cdot {}^{n-1}_n\mathbf{T} \quad (3.3)$$

3.3 Exemplu de calcul

Considerăm robotul de tip manipulator cu 4 grade de mobilitate analizat în cadrul primei lucrări (robot tip SCARA), căruia i se atașează sistemele de axe corespunzătoare fiecărui element cinematic, Fig. 3.2.

Parametrii Denavit-Hartenberg sunt prezentați în Tabelul 3.1. Acești parametri permit determinarea poziției punctului caracteristic M , ce coincide cu originea O_4 a sistemului de axe $x_6y_6z_6$, în raport cu originea O_0 a sistemului de referință $x_0y_0z_0$.

Mai întâi, se scriu matricele de transformare omogene corespunzătoare fiecărui element, la trecerea de la un sistem de axe la următorul, folosind relația (3.3). Se pleacă de la sistemul de referință $x_0y_0z_0$ spre sistemul de axe atașat punctului caracteristic M , respectiv $x_6y_6z_6$.

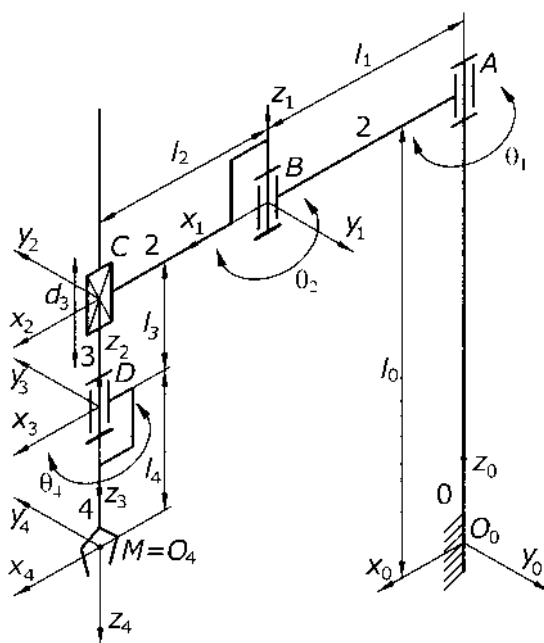


Fig. 3.2 Manipulator cu 4 grade de mobilitate (structură tip SCARA)

Tabelul 3.1 Parametrii Denavit - Hartenberg

	a_i	α_i	d_i	θ_i
1	l_1	0	l_0	θ_1
2	l_2	π	0	θ_2
3	0	0	$l_3 + d_3$	0
4	0	0	l_4	θ_4

$${}^0_1T = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 & l_1 \cdot \cos \theta_1 \\ \sin \theta_1 & \cos \theta_1 & 0 & l_1 \cdot \sin \theta_1 \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

$${}^1_2T = \begin{bmatrix} \cos \theta_2 & \sin \theta_2 & 0 & l_2 \cdot \cos \theta_2 \\ \sin \theta_2 & -\cos \theta_2 & 0 & l_2 \cdot \sin \theta_2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

$${}^2_3T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_3 + d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.6)$$

$${}^3_4T = \begin{bmatrix} \cos \theta_4 & -\sin \theta_4 & 0 & 0 \\ \sin \theta_4 & \cos \theta_4 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.7)$$

În final, se va determina matricea de transformare omogenă pentru întregul manipulator, aplicând relația (3.3).

$${}^0_4T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T = \begin{bmatrix} \cos(\theta_1 + \theta_2 - \theta_4) & \sin(\theta_1 + \theta_2 - \theta_4) & 0 & l_1 \cdot \cos \theta_1 + l_2 \cdot \cos(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2 - \theta_4) & -\cos(\theta_1 + \theta_2 - \theta_4) & 0 & l_1 \cdot \sin \theta_1 + l_2 \cdot \sin(\theta_1 + \theta_2) \\ 0 & 0 & -1 & l_0 - l_3 - d_3 - l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

Primele trei elemente ale primelor trei coloane din matricea (3.8) definesc orientarea obiectului manipulat în spațiul tridimensional, iar primele trei elemente ale coloanei a patra reprezintă coordonatele punctului caracteristic M al obiectului manipulat, în raport cu originea O_0 a sistemului de referință $x_0y_0z_0$.

3.4 Modul de lucru

- Pentru analiza cinematică directă se va folosi un manipulator din dotarea laboratorului, specificat de către conducătorul de lucrări, același manipulator care a fost utilizat și pentru lucrările anterioare;
- Se atașează mai întâi sistemele de axe corespunzătoare fiecărui element, conform paragrafelor 3.2 și 3.3;
- Se determină parametrii Denavit-Hartenberg și se trec într-un tabel, conform paragrafelor 3.2 și 3.3;
- Se scriu matricele de transformare omogene, la trecerea de la un sistem de axe la următorul;
- Se determină matricea totală de transformare omogenă a manipulatorului, în raport cu originea O_0 a sistemului de referință atașat elementului fix.

LUCRAREA 4

SIMULAREA PE CALCULATOR A CINEMATII DIRECTE A POZIȚIILOR

4.1 Scopul lucrării

Scopul acestei lucrări este de a efectua, cu ajutorul calculatorului, analiza cinematică directă a pozițiilor pentru un robot spațial de tip manipulator, folosind *parametrii Denavit-Hartenberg (D-H) standard*. Utilizând parametrii D-H ai manipulatorului, determinați în cadrul lucrării practice precedente, se va efectua calculul simbolic, respectiv un exemplu de calcul numeric al matricei omogene a acestuia.

4.2 Generalități

Pentru efectuarea acestei lucrări se folosește același manipulator, pentru care s-a efectuat analiza cinematică (Lucrarea 3), vezi Fig. 4.1.

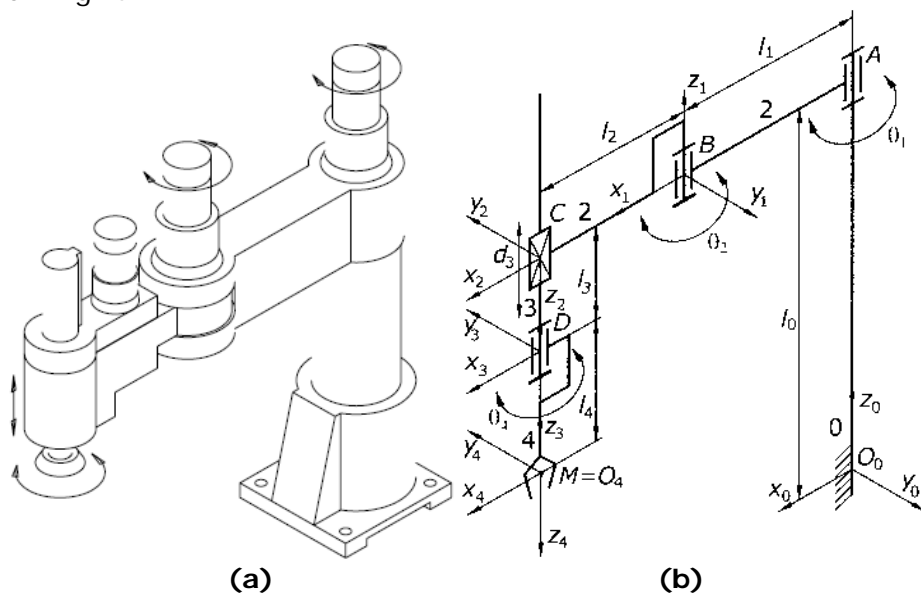


Fig. 4.1 Robot tip SCARA: a) model CAD; b) schemă cinematică

Robotică - Îndrumar de lucrări practice

Parametrii constructivi și cinematici ai manipulatorului sunt prezentați în Tabelul 4.1.

Tabelul 4.1

l_0 [mm]	l_1 [mm]	l_2 [mm]	l_3 [mm]	l_4 [mm]	θ_1 [°]	θ_2 [°]
600	400	300	0	150	± 120	± 132.5
d_3 [mm]	θ_4 [°]	ω_1 [°/s]	ω_2 [°/s]	v_3 [mm/s]	ω_4 [°/s]	
0 ÷ 200	± 180	112.5	114.5	200	225	

4.3 Program de calcul

% Program pentru analiza cinematica directa folosind calculul simbolic in % %
Matlab pentru un robot de tip Scara RRTR

%

clear;

clc;

% Parametrii D - H sunt stabiliti prin conventia D - H standard

%

%

syms a alfa d teta t1 t2 d3 t4 L0 L1 L2 L3 L4

% ti = unghiul de rotatie teta_i in cupla de rotatie

% di = deplasarea d_i in cupla de translatie

%

%

Ti_jat = [1 0 0 0; 0 1 0 0; 0 0 1 d; 0 0 0 1];

Ti_jar = [cos(teta) -sin(teta) 0 0; sin(teta) cos(teta) 0 0; 0 0 1 0; 0 0 0 1];

Ti_ja = Ti_jat*Ti_jar;

Ti_jbt = [1 0 0 a; 0 1 0 0; 0 0 1 0; 0 0 0 1];

Ti_jbr = [1 0 0 0; 0 cos(alfa) -sin(alfa) 0; 0 sin(alfa) cos(alfa) 0; 0 0 0 1];

Ti_jb = Ti_jbt*Ti_jbr;

%

% Matricea generala pentru transformarea omogena intre doua sisteme

%

Ti_j = Ti_ja*Ti_jb;

%

%

T0_1 = subs(Ti_j,{a,alfa,d,teta},{L1,0,L0,t1});

T1_2 = subs(Ti_j,{a,alfa,d,teta},{L2,pi,0,t2});

T2_3 = subs(Ti_j,{a,alfa,d,teta},{0,0,L3+d3,0});

T3_4 = subs(Ti_j,{a,alfa,d,teta},{0,0,L4,t4});

%

%

T0_2 = T0_1*T1_2;

Robotică - Îndrumar de lucrări practice

```

T0_3 = T0_1*T1_2*T2_3;
T0_4 = T0_1*T1_2*T2_3*T3_4;
%
%
T0_4
%
%
xM = simple(T0_4(1,4));
yM = simple(T0_4(2,4));
zM = simple(T0_4(3,4));
%
%
xM, yM, zM
%
%
% pause
% Calculul numeric pentru determinarea matricei omogene a manipulatorului %
si a coordonatelor punctului caracteristic M
%
%
% Determinarea matricei omogene a manipulatorului corespunzatoare pozitiei
% initiale
%
T0_4i=subs(T0_4,{L0,L1,L2,L3,L4,d3,t1,t2,t4},{0.6,0.4,0.3,0,0.15,0,0,0,0});
T0_4i
%
% Determinarea coordonatelor punctului caracteristic corespunzatoare pozitiei
% initiale
%
xMi = subs(xM,{L1,L2,t1,t2},{0.4,0.3,0,0});
yMi = subs(yM,{L1,L2,t1,t2},{0.4,0.3,0,0});
zMi = subs(zM,{L0,L3,L4,d3},{0.6,0,0.15,0});
%
%
fprintf('\t xMi = %12.6f \n\t yMi = %12.6f \n\t zMi = %12.6f \n',xMi,yMi,zMi);
%
% Determinarea matricei omogene a manipulatorului corespunzatoare pozitiei
% finale
%
T0_4f=subs(T0_4,{L0,L1,L2,L3,L4,d3,t1,t2,t4},{0.6,0.4,0.3,0,0.15,0.2,pi/3,pi/3,pi}
);
T0_4f
%
% Determinarea coordonatelor punctului caracteristic corespunzatoare pozitiei
% finale
%

```

```
xMf = subs(xM,{L1,L2,t1,t2},{0.4,0.3,pi/6,pi/3});  
yMf = subs(yM,{L1,L2,t1,t2},{0.4,0.3,pi/6,pi/3});  
zMf = subs(zM,{L0,L3,L4,d3},{0.6,0,0.15,0.2});  
%  
fprintf('\t xMf = %12.6f \n\t yMf = %12.6f \n\t zMf = %12.6f \n',xMf,yMf,zMf);  
%
```

4.4 Modul de lucru

- Pentru efectuarea acestei lucrări se va folosi același manipulator, care a fost utilizat și pentru lucrările anterioare;

- Plecând de la exemplul prezentat anterior, se scrie programul de calcul corespunzător robotului studiat, pentru determinarea simbolică a matricei omogene a manipulatorului și, respectiv, a relațiilor ce exprimă coordonatele punctului caracteristic M în raport cu sistemul de referință $\{0\}$;

- Se lansează programul, se salvează rezultatele și se compară cu cele obținute în cadrul lucrării practice 3;

- Se dau valori parametrilor cinematici de poziție ai cuplelor (în exemplul considerat, valori din Tabelul 4.1), pentru a efectua calculul numeric al matricei totale de transformare omogenă a manipulatorului, în raport cu originea O_0 a sistemului de referință atașat elementului fix, respectiv pentru a determina coordonatele punctului caracteristic, corespunzătoare parametrilor cinematici respectivi:

- mai întâi, se dau valori considerate inițiale, corespunzătoare poziției reprezentată în schema structurală (poziție pentru care, de regulă, parametrii cinematici de poziție ai cuplelor au valoarea zero); aceste valori ne vor conduce la coordonatele inițiale ale efectorului final (punctului caracteristic M);
- apoi, se dau valorile finale parametrilor cinematici de poziție, care vor conduce la deplasarea punctului caracteristic în poziția finală; aceste valori vor permite obținerea coordonatelor finale ale punctului caracteristic M ;

- Se notează valorile inițiale și finale ale parametrilor cinematici de poziție ai cuplelor, respectiv coordonatele inițiale și finale ale punctului caracteristic M . Acestea vor fi utilizate în *Lucrarea 5*.

LUCRAREA 5

ANALIZA CINEMATICALĂ INVERSĂ A POZIȚIILOR PENTRU ROBOȚII SPAȚIALI DE TIP MANIPULATOR

5.1 Scopul lucrării

Scopul acestei lucrări este acela de a efectua analiza cinematică inversă a pozițiilor pentru un robot spațial de tip manipulator din dotarea laboratorului, în vederea controlului mișcării acestuia în cadrul unor lucrări practice ulterioare.

5.2 Baze teoretice

5.2.1 Generalități

În practică, de cele mai multe ori se impune traiectoria pe care trebuie să o execute efectorul final și se pune problema determinării parametrilor cinematici ai cuplelor conducătoare, astfel încât să poată fi realizat controlul mișcării robotului. Altfel spus, se cunoaște poziția (coordonatele și orientarea) obiectului manipulat și se cer parametrii cinematici ai cuplelor manipulatorului (parametrii variabili θ_i, d_i), astfel încât să se asigure acea poziție impusă.

Problema cinematicii inverse este mult mai complexă, din următoarele motive:

- Ecuațiile ce trebuie rezolvate sunt, în general, neliniare, motiv pentru care nu este posibil întotdeauna să se găsească soluții distincte.
- Pot exista mai multe soluții (Fig. 5.1).
- Poate exista un număr infinit de soluții, cum ar fi cazul roboților redundanți.
- Se pot obține soluții inadmisibile din punct de vedere al geometriei robotului.

În ceea ce privește existența soluțiilor, acest lucru se poate garanta dacă poziția și orientarea efectorului final sunt din spațiul de lucru admisibil al robotului.

Pe de altă parte, problema soluțiilor multiple nu depinde numai de numărul gradelor de mobilitate ale robotului, ci și de numărul de parametri Denavit-Hartenberg nenuli. În general, cu cât numărul

parametrilor D-H nenuli este mai mare, cu atât numărul soluțiilor admisibile crește. Pentru un manipulator cu șase grade de mobilitate, fără limite mecanice pentru cuplele cinematice, există până la 16 soluții admisibile. Acest lucru necesită o serie de criterii de alegere a soluțiilor practice. Existența limitelor mecanice pentru cuple poate reduce numărul de soluții admisibile multiple.

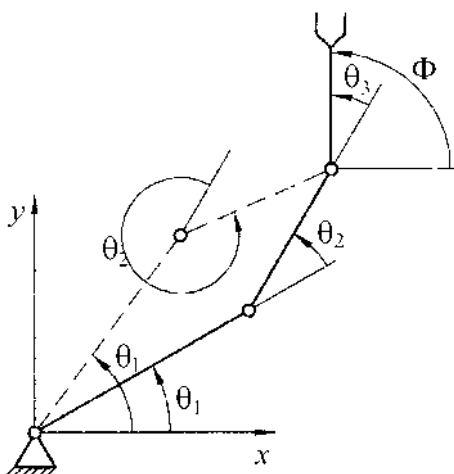


Fig. 5.1 Existența unor soluții multiple

Determinarea soluțiilor distincte necesită fie o intuiție algebrică, în vederea găsirii ecuațiilor semnificative ce conțin necunoscutele, fie o intuiție geometrică, pentru a găsi punctele semnificative ale structurii, în raport cu care este convenabil a se exprima poziția și/sau orientarea, ca o funcție de un număr redus de necunoscute. Atunci când nu există – sau este dificil să se găsească – soluții distincte, se poate apela la metode de rezolvare numerică. Aceste metode au avantajul că pot fi aplicate la orice structură cinematică, dar în general nu permit determinarea tuturor soluțiilor admisibile.

5.2.2 Considerații matematice preliminare

După cum se știe, funcțiile trigonometrice inverse au două valori în intervalul $(0, 360^\circ)$. Considerăm pentru exemplificare funcția:

$$y = \sin x \quad (5.1)$$

Robotică - Îndrumar de lucrări practice

Inversa funcției \sin are două valori în intervalul menționat, și anume:

$$\arcsin y = x \text{ și } \arcsin y = \pi - x \quad (5.2)$$

Similar, funcțiile inverse pentru \cos și \tan au, de asemenea, două valori în intervalul $(0, 360^\circ)$. Dacă:

$$y = \cos x \text{ sau } y = \tan x \quad (5.3)$$

atunci

$$\arccos y = x \text{ și } \arccos y = -x \quad (5.4)$$

respectiv

$$\arctan y = x \text{ și } \arctan y = \pi + x \quad (5.5)$$

Acest lucru crează probleme în ceea ce privește controlul robotului, prin faptul că există mai multe căi de a atinge poziția impusă efectorului final. Pentru evitarea acestor probleme, a fost definită funcția atan2 , care necesită două argumente și întoarce un singur răspuns în intervalul $(0, 360^\circ)$. Astfel, dacă vom considera funcțiile $s = \sin x$ și $c = \cos x$, funcția atan2 folosește ca argumente valorile s , c și returnează rezultatul x .

$$\text{atan2}(s, c) = x \quad (5.6)$$

Aceasta este implementată ca o funcție standard în limbaje de programare ca: C, Matlab, Pascal, Fortran, etc.

De asemenea, utilizând funcția inversă \arctg , atan2 poate fi definită și astfel:

$$\text{atan2}(s, c) = \begin{cases} \arctg\left(\frac{s}{c}\right), & \text{pentru } c > 0 \\ \arctg\left(\frac{s}{c}\right) + \pi \cdot \text{sign}(s), & \text{pentru } c < 0 \\ \frac{\pi}{2} \cdot \text{sign}(s), & \text{pentru } c = 0 \end{cases} \quad (5.7)$$

Pe de altă parte, în rezolvarea cinematicii inverse întâlnim frecvent ecuații de forma:

$$a \cdot \cos \alpha + b \cdot \sin \alpha + c = 0 \quad (5.8)$$

Pentru rezolvarea acestei ecuații se definește unghiul β , astfel încât:

$$\cos \beta = \frac{a}{\sqrt{a^2 + b^2}} \text{ și } \sin \beta = \frac{b}{\sqrt{a^2 + b^2}} \quad (5.9)$$

Trebuie precizat faptul că relațiile (5.9) pot fi scrise întotdeauna pentru ecuațiile de forma menționată. Unghiul β poate fi determinat utilizând funcția atan2 :

$$\beta = \text{atan2}\left(\frac{b}{\sqrt{a^2 + b^2}}, \frac{a}{\sqrt{a^2 + b^2}}\right) \quad (5.10)$$

În aceste condiții, ecuația noastră poate fi rescrisă astfel:

$$\sqrt{a^2 + b^2} \cdot \cos \beta \cdot \cos \alpha + \sqrt{a^2 + b^2} \cdot \sin \beta \cdot \sin \alpha + c = 0 \quad (5.11)$$

respectiv

$$\cos \beta \cdot \cos \alpha + \sin \beta \cdot \sin \alpha + \frac{c}{\sqrt{a^2 + b^2}} = 0 \quad (5.12)$$

sau

$$\cos(\alpha - \beta) = -\frac{c}{\sqrt{a^2 + b^2}} \quad (5.13)$$

Vom avea două soluții ale lui α , în funcție de β cunoscut:

$$\alpha = \beta + \arccos\left(\frac{-c}{\sqrt{a^2 + b^2}}\right) \text{ și } \alpha = \beta - \arccos\left(\frac{-c}{\sqrt{a^2 + b^2}}\right) \quad (6.14)$$

5.3 Exemplu

Considerăm robotul de tip manipulator cu 3 grade de mobilitate din Fig. 5.2, pentru care se presupune că analiza cinematică directă a fost deja efectuată. Parametrii Denavit-Hartenberg sunt prezentați în Tabelul 5.1.

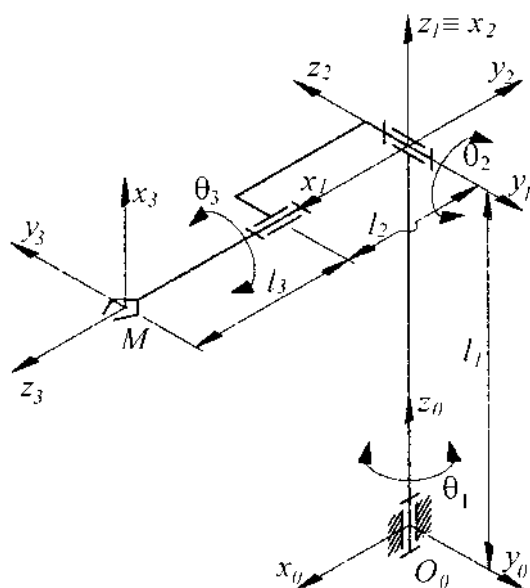


Fig. 5.2 Manipulator cu 3 grade de mobilitate

Tabelul 5.1 Parametrii Denavit - Hartenberg

	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	l_1	θ_1
2	90^0	0	0	$90^0 + \theta_2$
3	90^0	0	$l_2 + l_3$	θ_3

Elementele matricei omogene de transformare pentru întregul manipulator, ce definește poziția și orientarea efectorului final în raport cu originea O_0 a sistemului de referință $x_0y_0z_0$ se consideră cunoscută. Această matrice este de forma:

$${}^0_3T = \begin{pmatrix} a_{11} & a_{12} & a_{13} & x_M \\ a_{21} & a_{22} & a_{23} & y_M \\ a_{31} & a_{32} & a_{33} & z_M \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} -c_1s_2c_3 + s_1s_3 & c_1s_2s_3 + s_1c_3 & c_1c_2 & (l_2 + l_3)c_1c_2 \\ -s_1s_2c_3 - c_1s_3 & s_1s_2s_3 - c_1c_3 & s_1c_2 & (l_2 + l_3)s_1c_2 \\ c_2c_3 & -c_2s_3 & s_2 & (l_2 + l_3)s_2 + l_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.15)$$

unde:

Robotică - Îndrumar de lucrări practice

$$S_i = \sin(\theta_i)$$

$$C_i = \cos(\theta_i)$$

Elementele $a_{11}.....a_{33}$ ale matricei (5.15) definesc orientarea obiectului manipulat în spațiul tridimensional, iar elementele x_M, y_M, z_M reprezintă coordonatele punctului caracteristic M al obiectului manipulat, în raport cu originea O_0 a sistemului de referință $x_0y_0z_0$.

Grupând în mod convenabil elementele matricei omogene, se vor determina parametrii cinematici $\theta_1, \theta_2, \theta_3$ ai cuplelor.

Considerând elementele:

$$\begin{cases} x_M = (l_2 + l_3) \cdot \cos \theta_1 \cdot \cos \theta_2 \\ y_M = (l_2 + l_3) \cdot \sin \theta_1 \cdot \cos \theta_2 \end{cases} \quad (5.16)$$

se obține

$$\begin{cases} \cos \theta_1 = \frac{x_M}{(l_2 + l_3) \cdot \cos \theta_2} \\ \sin \theta_1 = \frac{y_M}{(l_2 + l_3) \cdot \cos \theta_2} \end{cases} \quad (5.17)$$

respectiv

$$\tan \theta_1 = \frac{y_M}{x_M} \quad (5.18)$$

Folosind funcția atan2 , obținem:

$$\theta_1 = \text{atan2}(y_M, x_M) \quad (5.19)$$

Considerăm apoi elementele:

$$\begin{cases} a_{31} = \cos \theta_2 \cdot \cos \theta_3 \\ a_{32} = -\cos \theta_2 \cdot \sin \theta_3 \\ z_M = (l_2 + l_3) \cdot \sin \theta_2 + l_1 \end{cases} \quad (5.20)$$

Ridicând la patrat primele două ecuații (5.20) și adunându-le, rezultă:

$$\begin{cases} \cos \theta_2 = \pm \sqrt{a_{31}^2 + a_{32}^2} \\ \sin \theta_2 = \frac{z_M - l_1}{l_2 + l_3} \end{cases} \quad (5.21)$$

Alegând sensul pozitiv pentru $\cos \theta_2$, se limitează valorile lui θ_2 în intervalul $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$, rezultând

$$\tan \theta_2 = \frac{\frac{z_M - l_1}{l_2 + l_3}}{\sqrt{a_{31}^2 + a_{32}^2}} \quad (5.22)$$

Folosind aceeași funcție *atan2*, obținem parametrul cinematic al celei de a doua cuplă:

$$\theta_2 = \text{atan2}\left(\frac{z_M - l_1}{l_2 + l_3}, \sqrt{a_{31}^2 + a_{32}^2}\right) \quad (5.23)$$

Similar, considerând elementele:

$$\begin{cases} a_{31} = \cos \theta_2 \cdot \cos \theta_3 \\ a_{32} = -\cos \theta_2 \cdot \sin \theta_3 \end{cases} \quad (5.24)$$

vom avea

$$\tan \theta_3 = \frac{-a_{32}}{a_{31}} \quad (5.25)$$

respectiv

$$\theta_3 = \text{atan2}(-a_{32}, a_{31}) \quad (5.26)$$

Pentru $\cos \theta_2 = -\sqrt{a_{31}^2 + a_{32}^2}$, adică lui θ_2 aparținând intervalului $\left(\frac{\pi}{2}, \frac{3\pi}{2}\right)$, avem:

$$\theta_1 = \text{atan2}(-y_M, -x_M) \quad (5.27)$$

$$\theta_2 = \text{atan2}\left(\frac{z_M - l_1}{l_2 + l_3}, -\sqrt{a_{31}^2 + a_{32}^2}\right) \quad (5.28)$$

$$\theta_3 = \text{atan2}(a_{32}, -a_{31}) \quad (5.29)$$

5.4 Modul de lucru

- Conducătorul de lucrări va preciza manipulatorul pentru care se va efectua analiza cinematică inversă;
- Urmărind etapele de lucru de la exemplul prezentat și ținându-se seama de recomandările făcute la paragraful 5.2.2, se vor determina parametrii cuplelor cinematice din structura manipulatorului precizat.

LUCRAREA 6

COMANDA ÎN POZIȚIE A EFECTORULUI FINAL PE BAZA REZULTATELOR CINEMATICEI DIRECTE

6.1 Scopul lucrării

Scopul acestei lucrări este de a comanda deplasarea efectorului final într-o anumită poziție, plecând de la o poziție inițială a acestuia, pe baza rezultatelor obținute la analiza cinematică inversă a pozițiilor, analiză efectuată folosind *parametrii Denavit-Hartenberg (D-H) standard*.

6.2 Generalități

Pentru exemplificare, considerăm același manipulator de tip SCARA, pentru care s-a efectuat analiza cinematică, Fig. 6.1.

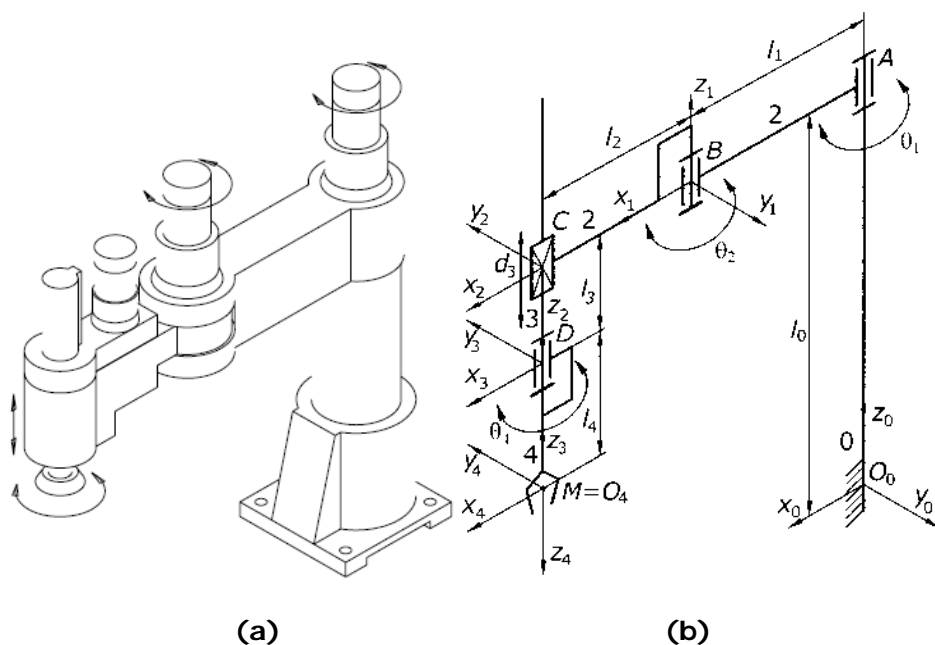


Fig. 6.1 Robot tip SCARA: a) model CAD; b) schemă cinematică

Parametrii constructivi și cinematici ai manipulatorului sunt prezentați în Tabelul 6.1.

Tabelul 6.1

l_0 [mm]	l_1 [mm]	l_2 [mm]	l_3 [mm]	l_4 [mm]	θ_1 [°]	θ_2 [°]
600	400	300	0	150	± 120	± 132.5
d_3 [mm]	θ_4 [°]	ω_1 [°/s]	ω_2 [°/s]	v_3 [mm/s]	ω_4 [°/s]	
0÷200	± 180	112.5	114.5	200	225	

6.3 Modul de lucru

- Pentru efectuarea acestei lucrări se va folosi același manipulator, care a fost utilizat și pentru lucrările anterioare;
- Plecând de la exemplul prezentat anterior, se scrie programul de comanda deplasarea efectorului final într-o anumită poziție, plecând de la o poziție inițială a acestuia, pe baza rezultatelor obținute la analiza cinematică inversă a pozițiilor.

LUCRAREA 7

CALCULUL MATRICEI JACOBIENE GEOMETRICE A UNUI ROBOT DE TIP MANIPULATOR

7.1 Scopul lucrării

Scopul acestei lucrări este de a determina matricea Jacobiană geometrică pentru un robot de tip manipulator cu structură serială.

7.2 Baze teoretice

7.2.1 Generalități

Atunci când se controlează deplasarea unui robot dintr-o anumită poziție în alta, nu este suficientă determinarea coordonatelor cuplelor cinematice și ale efectorului final, coordonate corespunzătoare poziției ce trebuie atinsă. În plus, uneori poate fi necesar controlul continuu al traiectoriei robotului, pe durata deplasării acestuia spre poziția cerută, în scopul evitării obstacolelor. De asemenea, există cazuri când efectorul final trebuie să urmărească precis o anumită traiectorie. Spre exemplu, în cazul operației de sudare este necesară menținerea electrodului la o anumită distanță (constantă) și cu o anumită orientare față de piesă, pe toată durata efectuării operației respective. Acest lucru necesită controlul vitezei efectorului final sau al sculei. Deoarece controlul mișcării robotului se realizează prin controlul cuplelor cinematice, viteza efectorului final va fi controlată efectuând un control al vitezelor acestor cuple. În acest sens, cunoscând viteza impusă efectorului final, este necesară determinarea vitezei fiecărei cuple cinematice.

Analiza cinematică a vitezelor oferă legătura dintre vitezele cuplelor cinematice și vitezele unghiulară și liniară corespunzătoare efectorului final. Acest lucru este descris de o matrice, numită *matrice Jacobiană geometrică*, care depinde de configurația manipulatorului. Alternativ, dacă este exprimată localizarea efectorului final în raport cu un sistem de referință printr-un set de ecuații analitice, atunci este posibilă calcularea matricei Jacobiene prin derivarea directă a ecuațiilor respective, în raport cu variabilele cuplelor cinematice.

Matricea Jacobiană rezultată, numită *matrice Jacobiană analitică*, diferă în general de cea geometrică. Matricea Jacobiană constituie unul din cele mai importante instrumente de caracterizare a roboților de tip manipulator. Aceasta este utilă pentru găsirea configurațiilor singulare, analiza redundanței, determinarea algoritmilor de analiză cinematică inversă, descrierea relațiilor dintre forțele aplicate efectorului final și momentele de torsiune rezultate în cuple (statică) și pentru obținerea ecuațiilor dinamice de mișcare.

7.2.2 Matricea Jacobiană geometrică

7.2.2.1 Noțiuni introductive

Considerăm un manipulator cu n grade de mobilitate, a cărui cinematică directă poate fi exprimată prin matricea generală de forma:

$${}^0_n\mathbf{T}(\mathbf{q}) = \begin{bmatrix} {}^0_n\mathbf{R}(\mathbf{q}) & {}^0_n\mathbf{p}(\mathbf{q}) \\ \mathbf{0} & 1 \end{bmatrix} \quad (7.1)$$

unde $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$ este vectorul variabilelor cuplelor cinematice, $q_i = \theta_i$ pentru o cuplă cinematică de rotație și $q_i = d_i$ pentru o cuplă de translație. Atât poziția, cât și orientarea efectorului final, depind de aceste variabile.

Scopul analizei cinematice a vitezelor este de a găsi legătura dintre vitezele cuplelor și vitezele liniară și unghiulară ale efectorului final. Cu alte cuvinte, se cere determinarea expresiei vitezei liniare $\dot{\mathbf{p}}$ și unghiulare $\boldsymbol{\omega}$, ca funcții de vitezele $\dot{\mathbf{q}}$ ale cuplelor cinematice:

$$\dot{\mathbf{p}} = \mathbf{J}_p(\mathbf{q}) \cdot \dot{\mathbf{q}}, \quad (7.2)$$

$$\boldsymbol{\omega} = \mathbf{J}_o(\mathbf{q}) \cdot \dot{\mathbf{q}}. \quad (7.3)$$

În relația (7.2), \mathbf{J}_p este o submatrice $3 \times n$ a matricei Jacobiene, reprezentând contribuția vitezelor cuplelor cinematice $\dot{\mathbf{q}}$ la viteza liniară $\dot{\mathbf{p}}$ a efectorului final, iar în (7.3) \mathbf{J}_o este o submatrice $3 \times n$, reprezentând contribuția vitezelor $\dot{\mathbf{q}}$ la viteza unghiulară $\boldsymbol{\omega}$ a efectorului final. În formă compactă, relațiile (7.2) și (7.3) pot fi scrise astfel:

$$\mathbf{v} = \begin{bmatrix} \dot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \mathbf{J}(\mathbf{q}) \cdot \dot{\mathbf{q}}, \quad (7.4)$$

care reprezintă ecuația cinematicii directe a vitezelor manipulatorului. Matricea \mathbf{J} de dimensiune $6 \times n$ poartă numele de *matrice Jacobiană geometrică* a manipulatorului,

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_O \end{bmatrix}, \quad (7.5)$$

care este, în general, o funcție de variabilele cuplelor cinematice.

7.2.2.2 Calculul matricei Jacobiene geometrice

Considerăm matricea Jacobiană (7.5) a unui manipulator cu n cuple cinematice, care conține $2n$ vectori coloană cu dimensiunea 3×1 :

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_O \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{P1} & \dots & \mathbf{J}_{Pn} \\ \mathbf{J}_{O1} & \dots & \mathbf{J}_{On} \end{bmatrix}, \quad (7.6)$$

unde vectorii \mathbf{J}_{Pi} reprezintă contribuția cuplei cinematice i la viteza liniară a efectorului final, iar \mathbf{J}_{Oi} constituie contribuția cuplei cinematice i la viteza unghiulară a acestuia. Se poate demonstra că acești vectori au forma

$$\begin{bmatrix} \mathbf{J}_{Pi} \\ \mathbf{J}_{Oi} \end{bmatrix} = \begin{cases} \begin{bmatrix} \mathbf{z}_{i-1} \times \left({}^0_n\mathbf{p} - {}^{i-1}_i\mathbf{p} \right) \\ \mathbf{z}_{i-1} \end{bmatrix} & \text{pentru o cupla cinematică de } \textit{rotatie} \\ \begin{bmatrix} \mathbf{z}_{i-1} \\ \mathbf{0} \end{bmatrix} & \text{pentru o cupla cinematică de } \textit{translatie} \end{cases}, \quad (7.7)$$

pentru $i = 1..n$, unde:

- vectorul \mathbf{z}_{i-1} este reprezentat de a treia coloană a matricei de rotație ${}^{0}_{i-1}\mathbf{R}$ sau se poate calcula cu relația

$$\mathbf{z}_{i-1} = {}^{0}_{i-1}\mathbf{R} \cdot \mathbf{z}_0, \quad (7.8)$$

cu

$$\mathbf{z}_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}; \quad (7.9)$$

• vectorul ${}^0_{i-1}\mathbf{p}$ este reprezentat de primele trei elemente ale coloanei a patra din matricea omogenă ${}^0_{i-1}\mathbf{T}$ sau se poate calcula astfel

$${}^0_{i-1}\mathbf{p} = {}^0_{i-1}\mathbf{T} \cdot {}^0_0\mathbf{p}, \quad (7.10)$$

cu

$${}^0_0\mathbf{p} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}; \quad (7.11)$$

• vectorul ${}^0_n\mathbf{p}$ este reprezentat de primele trei elemente ale coloanei a patra din matricea omogenă ${}^0_n\mathbf{T}$ sau se poate calcula cu relația

$${}^0_n\mathbf{p} = {}^0_n\mathbf{T} \cdot {}^0_0\mathbf{p}. \quad (7.12)$$

Din ${}^0_0\mathbf{p}$, respectiv din produsele (7.10) și (7.12), se rețin numai primii trei termeni.

Cu alte cuvinte, matricea Jacobiană geometrică a unui manipulator cu n grade de mobilitate, având toate cele n cuple cinematice de rotație, capătă forma

$$\mathbf{J} = \begin{bmatrix} \mathbf{z}_0 \times \left({}^0_n\mathbf{p} - {}^0_0\mathbf{p} \right) & \cdots & \mathbf{z}_{i-1} \times \left({}^0_n\mathbf{p} - {}^0_{i-1}\mathbf{p} \right) & \cdots & \mathbf{z}_{n-1} \times \left({}^0_n\mathbf{p} - {}^0_{n-1}\mathbf{p} \right) \\ \mathbf{z}_0 & \cdots & \mathbf{z}_{i-1} & \cdots & \mathbf{z}_{n-1} \end{bmatrix}. \quad (7.13)$$

Pentru un manipulator cu n grade de mobilitate, având $n-1$ cuple cinematice de rotație și cupla cinematică i de translație, matricea Jacobiană are forma generală

$$\mathbf{J} = \begin{bmatrix} \mathbf{z}_0 \times \left({}^0_n\mathbf{p} - {}^0_0\mathbf{p} \right) & \cdots & \mathbf{z}_{i-1} & \cdots & \mathbf{z}_{n-1} \times \left({}^0_n\mathbf{p} - {}^0_{n-1}\mathbf{p} \right) \\ \mathbf{z}_0 & \cdots & \mathbf{0} & \cdots & \mathbf{z}_{n-1} \end{bmatrix}. \quad (7.14)$$

7.2.2.3 Schimbarea sistemului de referință al matricei Jacobiene

Trebuie remarcat faptul că ecuațiile scrise anterior pot fi folosite pentru a calcula vitezele liniare și unghiulare a oricărui punct din structura manipulatorului, atunci când se cunoaște funcția ce definește cinematica directă a punctului respectiv.

De asemenea, trebuie notat că matricea Jacobiană depinde de sistemul de axe în raport cu care au fost exprimate vitezele efectorului final. Ecuațiile prezentate anterior permit determinarea matricei Jacobiene în raport cu sistemul de referință fix $\{0\}$. Dacă se cere reprezentarea matricei Jacobiene în raport cu un alt sistem de axe, este suficient să cunoaștem matricea de rotație relativă scrisă la trecerea de la sistemul de axe respectiv la sistemul de referință $\{0\}$.

Fie ${}^0\mathbf{J}$ matricea Jacobiană a unui manipulator, scrisă în raport cu sistemul de referință $\{0\}$, permite determinarea vitezelor efectorului final în raport cu sistemul de referință menționat

$${}^0\mathbf{v} = \begin{bmatrix} {}^0\dot{\mathbf{p}} \\ {}^0\boldsymbol{\omega} \end{bmatrix} = {}^0\mathbf{J} \cdot \dot{\mathbf{q}}. \quad (7.15)$$

Presupunem că dorim să calculăm vitezele efectorului în raport cu alt sistem de axe, sistemul $\{a\}$ spre exemplu. Mai întâi, trebuie notat faptul că vectorul viteză ${}^0\mathbf{v}$ cu dimensiunea 6×1 este descris în raport cu sistemul $\{a\}$ de transformarea

$${}^a\mathbf{v} = \begin{bmatrix} {}^a\dot{\mathbf{p}} \\ {}^a\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} {}^a_0\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^a_0\mathbf{R} \end{bmatrix} \cdot \begin{bmatrix} {}^0\dot{\mathbf{p}} \\ {}^0\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} {}^a_0\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^a_0\mathbf{R} \end{bmatrix} \cdot {}^0\mathbf{v}. \quad (7.16)$$

Deci, putem scrie

$$\begin{bmatrix} {}^a\dot{\mathbf{p}} \\ {}^a\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} {}^a_0\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^a_0\mathbf{R} \end{bmatrix} \cdot {}^0\mathbf{J} \cdot \dot{\mathbf{q}}, \quad (7.17)$$

ceea ce înseamnă că schimbarea sistemului de referință a unei matrice Jacobiene este descrisă de următoarea relație

$${}^a\mathbf{J} = \begin{bmatrix} {}^a_0\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^a_0\mathbf{R} \end{bmatrix} \cdot {}^0\mathbf{J}, \quad (7.18)$$

7.3 Exemplu de calcul

Considerăm manipulatorul antropomorf din Fig. 7.1, pentru care matricele de rotație, respectiv matricele de transformare (convenția D-H modificată), sunt după cum urmează:

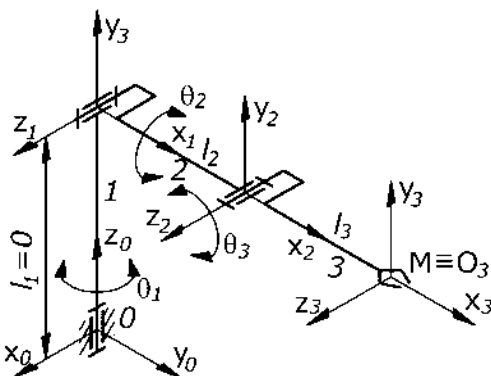


Fig.7.1 Manipulator antropomorf

$${}^0_1\mathbf{T} = \begin{bmatrix} -\sin\theta_1 & 0 & \cos\theta_1 & 0 \\ \cos\theta_1 & 0 & \sin\theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7.19)$$

$${}^1_2\mathbf{T} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & l_2 \cdot \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & l_2 \cdot \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7.20)$$

$${}^2_3\mathbf{T} = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & l_3 \cdot \cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & l_3 \cdot \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7.21)$$

$${}^0_2\mathbf{T} = {}^0_1\mathbf{T} \cdot {}^1_2\mathbf{T} = \begin{bmatrix} -s_1 \cdot c_2 & s_1 \cdot s_2 & c_1 & -l_2 \cdot s_1 \cdot c_2 \\ c_1 \cdot c_2 & -c_1 \cdot s_2 & s_1 & l_2 \cdot c_1 \cdot c_2 \\ s_2 & c_2 & 0 & l_2 \cdot s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7.22)$$

$${}^0_3T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T = \begin{bmatrix} -s_1 \cdot c_{23} & s_1 \cdot s_{23} & c_1 & -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ c_1 \cdot c_{23} & -c_1 \cdot s_{23} & s_1 & (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ s_{23} & c_{23} & 0 & l_2 \cdot s_2 + l_3 \cdot s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (7.23)$$

$${}^0_1R = \begin{bmatrix} -\sin \theta_1 & 0 & \cos \theta_1 \\ \cos \theta_1 & 0 & \sin \theta_1 \\ 0 & 1 & 0 \end{bmatrix}, \quad (7.24)$$

$${}^1_2R = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 \\ \sin \theta_2 & \cos \theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.25)$$

$${}^2_3R = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 \\ \sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.26)$$

$${}^0_2R = {}^0_1R \cdot {}^1_2R = \begin{bmatrix} -s_1 c_2 & s_1 s_2 & c_1 \\ c_1 c_2 & -c_1 s_2 & s_1 \\ s_2 & c_2 & 0 \end{bmatrix}. \quad (7.27)$$

În relațiile anterioare s-au făcut notațiile: $s_i = \sin \theta_i$, $c_i = \cos \theta_i$, $s_{ij} = \sin(\theta_i + \theta_j)$, $c_{ij} = \cos(\theta_i + \theta_j)$.

Matricea Jacobiană a manipulatorului va fi

$${}^0J = \begin{bmatrix} \mathbf{z}_0 \times \left({}^0_3\mathbf{p} - {}^0_0\mathbf{p} \right) & \mathbf{z}_1 \times \left({}^0_3\mathbf{p} - {}^0_1\mathbf{p} \right) & \mathbf{z}_2 \times \left({}^0_3\mathbf{p} - {}^0_2\mathbf{p} \right) \\ \mathbf{z}_0 & \mathbf{z}_1 & \mathbf{z}_2 \end{bmatrix}, \quad (7.28)$$

unde:

$$\mathbf{z}_1 = {}^0_1R \cdot \mathbf{z}_0 = \begin{bmatrix} -s_1 & 0 & c_1 \\ c_1 & 0 & s_1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 \\ s_1 \\ 0 \end{bmatrix}; \quad (7.29)$$

$$\mathbf{z}_2 = {}^0_2R \cdot \mathbf{z}_0 = \begin{bmatrix} -s_1 c_2 & s_1 s_2 & c_1 \\ c_1 c_2 & -c_1 s_2 & s_1 \\ s_2 & c_2 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 \\ s_1 \\ 0 \end{bmatrix}; \quad (7.30)$$

$${}^0_1\mathbf{T} \cdot {}^0_0\mathbf{p} = \begin{bmatrix} -s_1 & 0 & c_1 & 0 \\ c_1 & 0 & s_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad (7.31)$$

adică

$${}^0_1\mathbf{p} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad (7.32)$$

$${}^0_2\mathbf{T} \cdot {}^0_0\mathbf{p} = \begin{bmatrix} -s_1 \cdot c_2 & s_1 \cdot s_2 & c_1 & -l_2 \cdot s_1 \cdot c_2 \\ c_1 \cdot c_2 & -c_1 \cdot s_2 & s_1 & l_2 \cdot c_1 \cdot c_2 \\ s_2 & c_2 & 0 & l_2 \cdot s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -l_2 \cdot s_1 \cdot c_2 \\ l_2 \cdot c_1 \cdot c_2 \\ l_2 \cdot s_2 \\ 1 \end{bmatrix}, \quad (7.33)$$

adică

$${}^0_2\mathbf{p} = \begin{bmatrix} -l_2 \cdot s_1 \cdot c_2 \\ l_2 \cdot c_1 \cdot c_2 \\ l_2 \cdot s_2 \end{bmatrix}; \quad (7.34)$$

$${}^0_3\mathbf{T} \cdot {}^0_0\mathbf{p} = \begin{bmatrix} -s_1 c_{23} & s_1 s_{23} & c_1 & -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ c_1 c_{23} & -c_1 s_{23} & s_1 & (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ s_{23} & c_{23} & 0 & l_2 \cdot s_2 + l_3 \cdot s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ l_2 \cdot s_2 + l_3 \cdot s_{23} \\ 1 \end{bmatrix}, \quad (7.35)$$

adică

$${}^0_3\mathbf{p} = \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ l_2 \cdot s_2 + l_3 \cdot s_{23} \end{bmatrix}; \quad (7.36)$$

$${}^0_3\mathbf{p} - {}^0_0\mathbf{p} = \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ l_2 \cdot s_2 + l_3 \cdot s_{23} \end{bmatrix}; \quad (7.37)$$

$${}^0_3\mathbf{p} - {}^0_1\mathbf{p} = \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 \\ (l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 \\ l_2 \cdot s_2 + l_3 \cdot s_{23} \end{bmatrix}; \quad (7.38)$$

$${}^0_3\mathbf{p} - {}^0_2\mathbf{p} = \begin{bmatrix} -l_3 \cdot s_1 \cdot c_{23} \\ l_3 \cdot c_1 \cdot c_{23} \\ l_3 \cdot s_{23} \end{bmatrix}. \quad (7.39)$$

Matricea Jacobiană finală a manipulatorului, scrisă în raport cu sistemul de referință $\{0\}$, vezi relația (7.28), este

$${}^0\mathbf{J} = \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 & (l_2 \cdot s_2 + l_3 \cdot s_{23}) \cdot s_1 & l_3 \cdot s_1 \cdot s_{23} \\ -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 & -(l_2 \cdot s_2 + l_3 \cdot s_{23}) \cdot c_1 & -l_3 \cdot c_1 \cdot s_{23} \\ 0 & l_2 \cdot c_2 + l_3 \cdot c_{23} & l_3 \cdot c_{23} \\ 0 & c_1 & c_1 \\ 0 & s_1 & s_1 \\ 1 & 0 & 0 \end{bmatrix}. \quad (7.40)$$

Matricea Jacobiană geometrică raportată la sistemul de axe $\{3\}$, atașat efectorului final se calculează cu relația

$${}^3\mathbf{J} = \begin{bmatrix} {}^3_0\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}^3_0\mathbf{R} \end{bmatrix} \cdot {}^0\mathbf{J} =, \quad (7.41)$$

Conoscând matricele ${}^0_2\mathbf{R}$, relația (7.27), și ${}^2_3\mathbf{R}$ (7.26), putem determina matrice de rotație totală ${}^0_3\mathbf{R}$, ca fiind

$${}^0_3\mathbf{R} = {}^0_2\mathbf{R} \cdot {}^2_3\mathbf{R} = \begin{bmatrix} -s_1 c_2 & s_1 s_2 & c_1 \\ c_1 c_2 & -c_1 s_2 & s_1 \\ s_2 & c_2 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_1 c_{23} & s_1 s_{23} & c_1 \\ c_1 c_{23} & -c_1 s_{23} & s_1 \\ s_{23} & c_{23} & 0 \end{bmatrix}. \quad (7.42)$$

Pentru o matrice de rotație, inversa acesteia este egală cu transpusa ei. De aceea, matricea ${}^3_0\mathbf{R}$ va fi

$${}^3_0\mathbf{R} = {}^0_3\mathbf{R}^{-1} = {}^0_3\mathbf{R}^T = \begin{bmatrix} -s_1 c_{23} & c_1 c_{23} & s_{23} \\ s_1 s_{23} & -c_1 s_{23} & c_{23} \\ c_1 & s_1 & 0 \end{bmatrix}. \quad (7.43)$$

Revenind la ecuația (7.41), obținem:

$$\begin{aligned} {}^3\mathbf{J} &= \begin{bmatrix} -s_1 c_{23} & c_1 c_{23} & s_{23} & 0 & 0 & 0 \\ s_1 s_{23} & -c_1 s_{23} & c_{23} & 0 & 0 & 0 \\ c_1 & s_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -s_1 c_{23} & c_1 c_{23} & s_{23} \\ 0 & 0 & 0 & s_1 s_{23} & -c_1 s_{23} & c_{23} \\ 0 & 0 & 0 & c_1 & s_1 & 0 \end{bmatrix} \cdot \\ &\cdot \begin{bmatrix} -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot c_1 & (l_2 \cdot s_2 + l_3 \cdot s_{23}) \cdot s_1 & l_3 \cdot s_1 \cdot s_{23} \\ -(l_2 \cdot c_2 + l_3 \cdot c_{23}) \cdot s_1 & -(l_2 \cdot s_2 + l_3 \cdot s_{23}) \cdot c_1 & -l_3 \cdot c_1 \cdot s_{23} \\ 0 & l_2 \cdot c_2 + l_3 \cdot c_{23} & l_3 \cdot c_{23} \\ 0 & c_1 & c_1 \\ 0 & s_1 & s_1 \\ 1 & 0 & 0 \end{bmatrix} = \\ &= \begin{bmatrix} 0 & l_2 \cdot (s_2 + c_2) & \end{bmatrix}. \end{aligned} \quad (7.44)$$

7.4 Modul de lucru

- Studenții vor determina matricea Jacobiană simbolică, în raport cu sistemul de referință $\{0\}$, respectiv în raport cu sistemul de axe atașat efectorului final, pentru un manipulator din dotarea laboratorului, a cărei analiză cinematică a pozițiilor a fost efectuată în cadrul unei lucrări practice anterioare.

LUCRAREA 8

COMANDA UNUI SERVOMECHANISM DE CURENT CONTINUU, CU MICROCONTROLLER PIC16F84

8.1 Scopul lucrării

Scopul acestei lucrări este de a prezenta studenților componentele de bază ale unui servomecanism, structura generală a unui microcontroller PIC 16F84, precum și utilizarea acestuia pentru comanda poziției unui servomecanism de curent continuu.

8.2 Servomecanism de curent continuu

8.2.1 Prezentare generală

Acest tip de actuator are avantajul de a avea integrate în aceeași carcasă, pe lângă **motorul electric**, un **modul electronic** și un **modul mecanic** (Fig. 8.1).

Modulul electronic are un rol foarte important în funcționarea precisă a sistemului de acționare. Pe lângă componentele necesare prelucrării semnalelor de intrare și a celor de comandă, acesta conține și un traductor de poziție unghiulară de tip rezistiv (un potențiomtru). Acest traductor are un rol important în cinematica sistemului, el fiind responsabil atât de "citirea" poziției unghiulare a rotorului.

Datorită modulului de control digital, servomotorul este foarte des utilizat în construcția sistemelor electro-mecanice cu comandă numerică, și anume: în industria IT (copiatoare, CD-ROM-uri, HDD-uri etc), în industria prelucrării materialelor feroase și lemnoase (strunguri, freze, mașini de găurit cu comandă numerică), robotică (actuatori pentru roboți mobili pășitori, roboți de tip manipulator), etc.

Modulul mecanic constă dintr-un reductor, ce are rolul de a mări cuplul motorului în detrimentul vitezei de rotație. Foarte adesea, un motor are la ieșire o viteză unghiulară a rotorului foarte mare și poate produce destulă putere, însă nu destul cuplu, necesar sistemului de acționare.

Acest tip de actuator este folosit foarte des și în aeromodelism, datorită masei sale mici și a posibilității de control al poziției cu precizie ridicată.

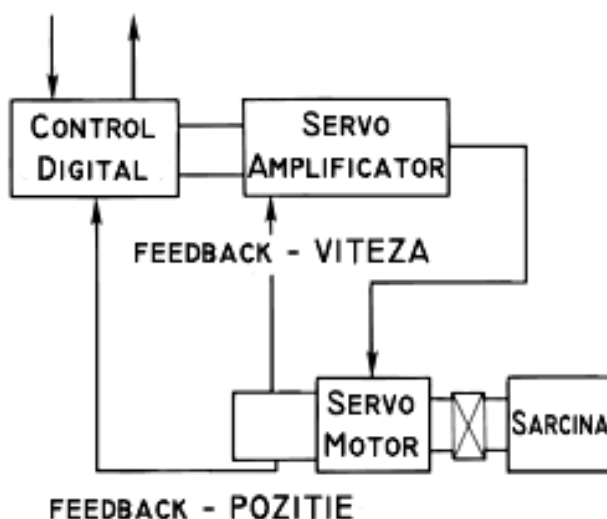


Fig. 8.1 Principiul de funcționare al unui servomecanism

Servomecanismele sunt motoreductoare de curent continuu, cu circuit de comandă în poziție, ce permite poziționarea cu precizie a axului de ieșire. Totuși, dacă momentul de torsiune rezistiv al sarcinii antrenate depășește o valoare limită, este posibil ca poziția unghiulară comandată să nu fie atinsă. Cursa unghiulară a axului este de 180° ($\pm 90^\circ$).

Componentele de bază ale unui servomecanism sunt prezentate în Fig. 8.2., în vedere explodată, iar caracteristicile tehnice ale servomecanismului ce va fi utilizat în această aplicație (Futaba S3001, vezi Fig. 8.3) sunt prezentate în Tabelul 8.1.

Tabelul 8.1

Caracteristică	Valoare
Tensiune de alimentare	4,8-6 [V]
Curentul în vid	8 [mA]
Curentul maxim	720 [mA]
Cuplul maxim	0,34 [Nm]
Cursa unghiulară	180°
Viteza unghiulară maximă	1,8 [rad/s]
Masa	45.1 [g]
Dimensiuni	41x35x20 [mm]

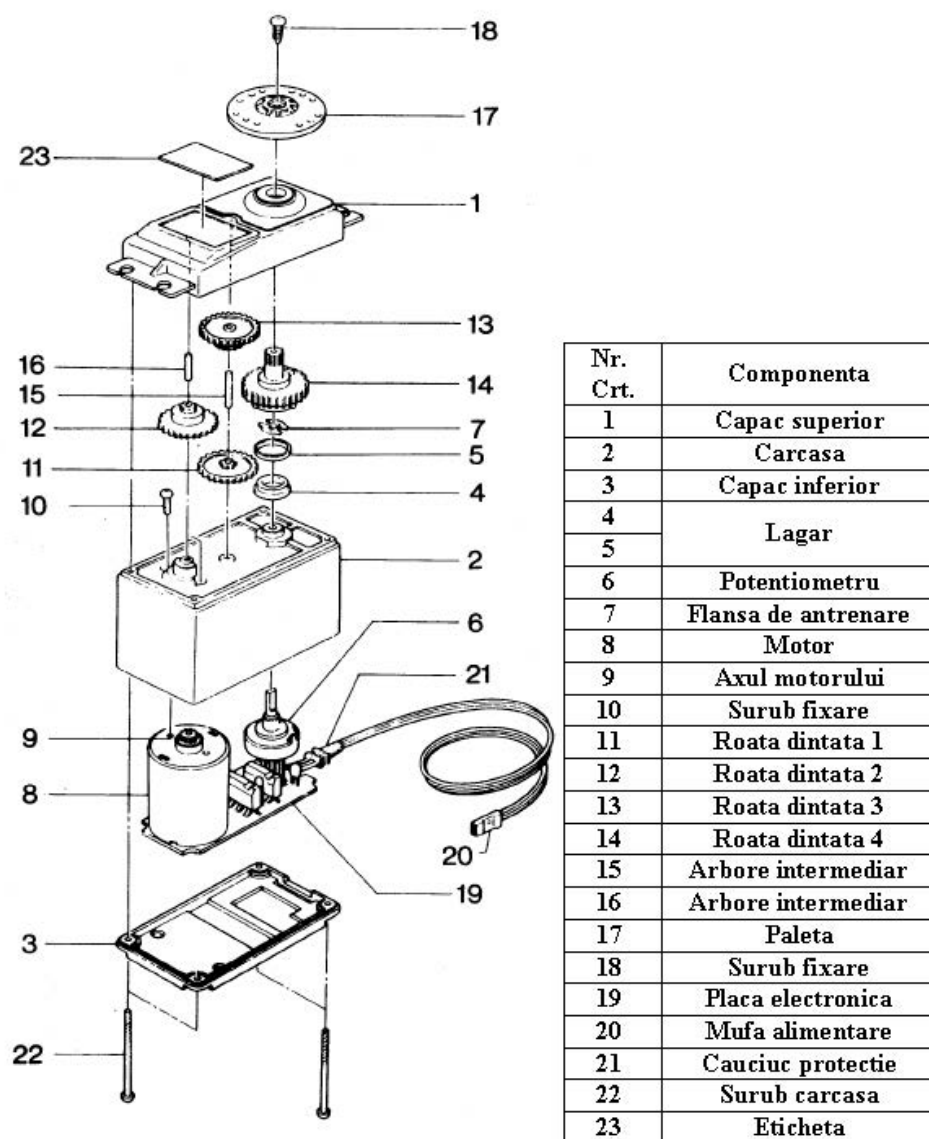


Fig. 8.2 Componentele unui servomecanism



Fig. 8.3 Servomecanism Futaba S3001

Motorul 8 este acționat prin impulsuri ce sunt emise de modulul electronic de control. Cuplul acestuia este amplificat de un reductor format din roțile dințate 11, 12, 13, 14 și arborii intermediari 15, 16.

Alimentarea și controlul se face prin intermediul conectorului 20. Toate componentele sunt înglobate în aceeași carcasă de mici dimensiuni (carcasa 2, plus capacele 1 și 3, prinse cu șuruburile 22).

Cablul de alimentare și de comanda al unui servomotor are trei fire: două pentru alimentare și al treilea pentru transmiterea semnalelor de comandă a poziției unghiulare. Semnalul de comandă este sub forma unui tren de impulsuri, a caror lățime poate varia între 1 și 2 [ms]. Mărimea impulsului determină poziția arborelui de ieșire al servomecanismului.

8.2.2 Principiul de control al poziției unui servomecanism

Un servomecanism este ideal pentru a controla un element cinematic aflat în mișcare de oscilație între două poziții, una de minim la 0° și una de maxim la 180° , trecând printr-o poziție de nul la 90° .

Limitarea deplasării se face mecanic, printr-un limitator atașat axului de ieșire al servomecanismului.

Comanda poziției arborelui de ieșire al servomecanismului se face prin impulsuri, cu o lățime ce variază între 1 și 2 [ms]. Acest tip de control se numește PCM (Pulse Cod Modulation). Lățimea impulsului va determina poziția unghiulară a axului.

Cu alte cuvinte, dacă poziția neutră (90°) este dată, teoretic, de un impuls cu lățimea de 1.5 [ms], atunci când acesta va fi mai mare de 1.5 [ms] axul se va roti către poziția de 180° , iar dacă va fi mai mic de această valoare, atunci axul se va roti către poziția de 0° (Fig. 8.4). Poziția unghiulară este direct proporțională cu lățimii impulsului, ceea ce înseamnă că, pentru a avea o poziționare la un unghi de 45° , lățimea impulsului trebuie să fie, teoretic, de 1.375 [ms]. În practică, aceste valori pot să fie puțin diferite, pentru servomecanisme diferite, datorită preciziei de montaj a componentelor mecanice, dar și a caracteristicii potențiometrului. De aceea, se recomandă o etalonare prealabilă a fiecărui servomecanism, înainte de utilizarea acestuia.

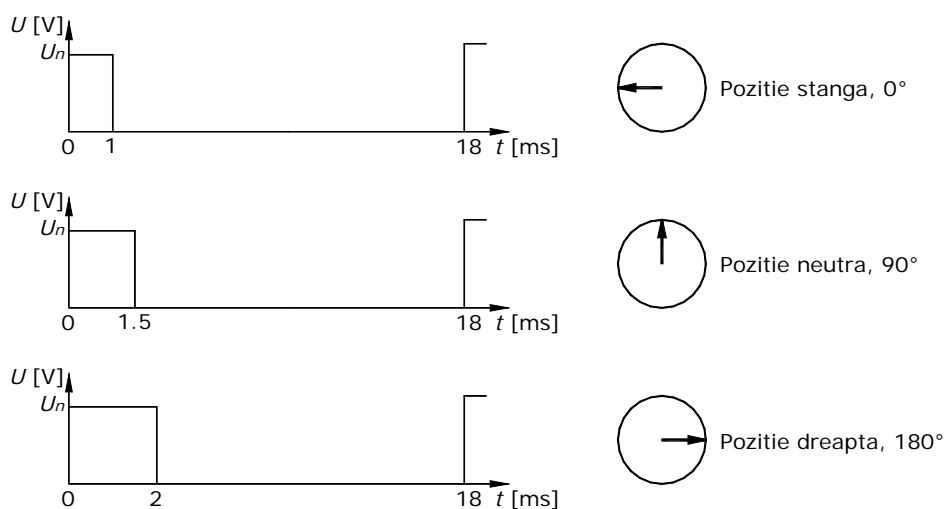


Fig. 8.4 Poziționarea axului funcție de lățimea impulsului

Perioada T recomandată a semnalului de comandă trebuie să fie cuprinsă între 18 și 20 [ms]. În practică, aceasta poate avea și alte valori. Drept consecință, lățimea t a impulsului va dicta poziția unghiulară, iar raportul t/T va determina viteza unghiulară a axului de ieșire.

8.3 Platforma de dezvoltare cu microcontroler PIC16F84

8.3.1 Generalități

Microcontrolerul, la modul general, diferă de un microprocesor în mai multe feluri. Primul aspect, și cel mai important, îl constituie funcționalitatea sa. Pentru a fi folosit, unui microprocesor trebuie să i se adauge alte componente cum ar fi memoria, sau componentele pentru primirea și transmiterea de date. Pe scurt, aceasta înseamnă că microprocesorul reprezintă inima calculatorului. Pe de altă parte, microcontrolerul este conceput astfel încât să conțină toate aceste componente într-un singur circuit integrat. Nu sunt necesare alte componente externe pentru utilizarea sa, deoarece toate perifericele sunt deja incluse în acesta. Astfel, economisim timp și spațiu, necesare construirii de dispozitive electronice.

Schema generală a unui microcontroler, cu elementele lui de bază și conexiunile interne, este prezentată în Fig. 8.5.

Memoria este o parte a microcontrolerului, a cărei funcție este de a înmagazina date. Cel mai simplu mod de a explica semnificația acesteia, este de a o considera ca fiind un dulap mare, cu multe sertare. Dacă presupunem că am marcat sertarele într-un anume mod, încât să nu fie confundate între ele, conținutul acestora va fi ușor accesibil. Este suficient să se cunoască sertarul desemnat și conținutul va fi cunoscut în mod sigur. Memoria constă dintr-o serie de locații de memorie.

Unitatea centrală de procesare (CPU) este constituită dintr-o serie de registre, ce reprezintă locații de memorie, al căror rol este de a ajuta prin executarea unor operații matematice variate. Conectarea memoriei și a unității centrale se realizează folosind „bus”-uri.

Unitatea de intrare/ieșire (I/O) permite comunicarea cu exteriorul, prin intermediul „porturilor”, care pot fi: de intrare, de ieșire sau cu două căi (intrare/ieșire).

Unitatea serială permite face legătura cu lumea externă, respectiv permite transmiterea de date din sau către exterior, folosind trei linii de comunicație.

Unitatea timer generează semnale la intervale egale de timp, reprezentând un contror de timp („clock”).

„Watchdog”-ul („câinele de pază”) întrerupe (resetează) programul atunci când microcontrolerul începe să funcționeze incorect sau intră într-o buclă infinită. De aceea, atunci când în program sunt

introduse în mod intenționat asemenea bucle, trebuie eliminată obținerea watchdog în faza de încărcare a programului.

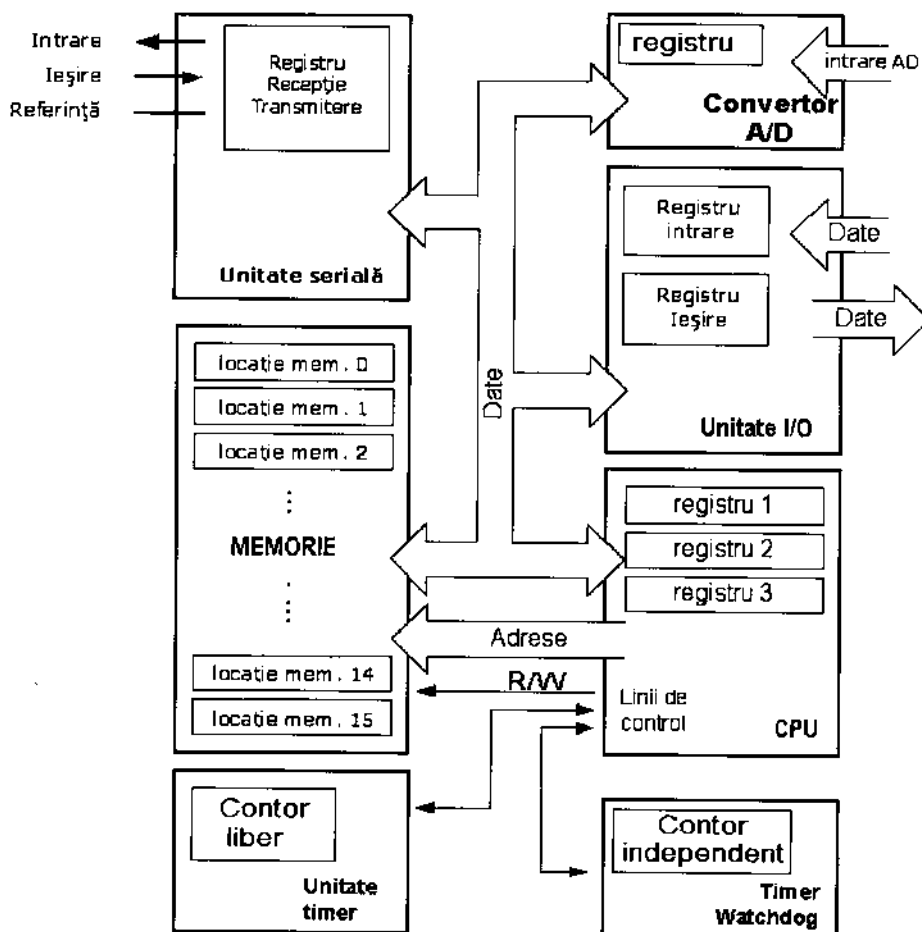


Fig. 8.5 Schema generală a unui microcontroler

Convertorul analog-digital (A/D) convertește un semnal analogic într-un număr binar (digital).

8.3.2 Microcontrolerul PIC16F84

PIC16F84 (Fig. 8.6) aparține unei clase de microcontrolere de 8 biți cu arhitectura RISC. Structura lui generală este arătată în Fig. 8.6, reprezentând blocurile de bază ale acestuia.

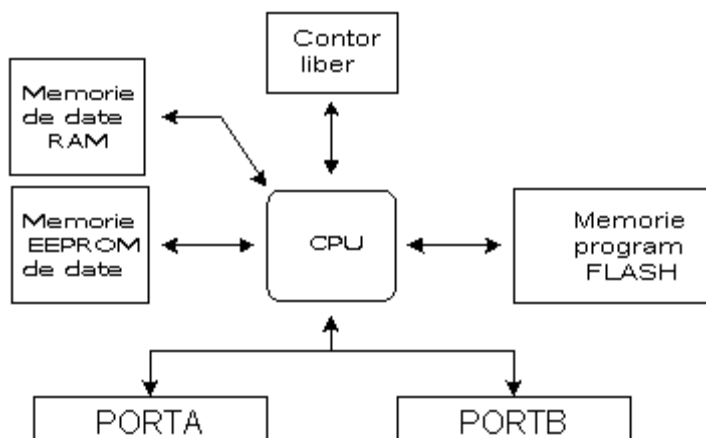
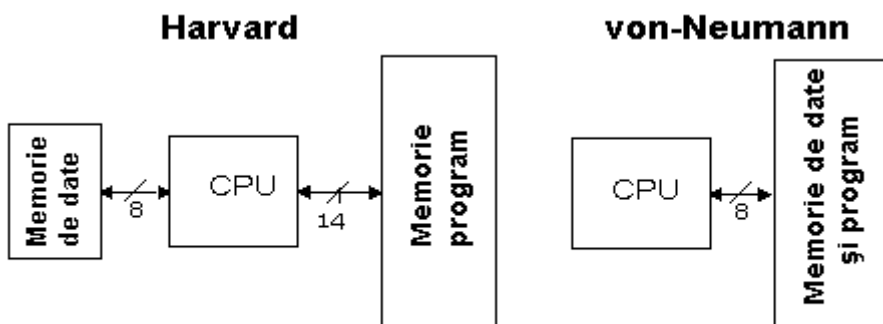


Fig. 8.6 Schița microcontrolerului PIC16F84

Microcontrolerele pot fi cu arhitectura Harvard sau von-Neuman (Fig. 8.7). Arhitectura Harvard este un concept mai nou decât von-Neuman, apărând din nevoia de a mări viteza microcontrolerului. În această arhitectură, bus-ul de date și bus-ul de adrese sunt separate, fiind posibil un debit mare de date prin unitatea centrală de procesare și o viteză mare de lucru. De asemenea, arhitectura Harvard folosește mai puține instrucțiuni decât von-Neuman.

Microcontrolerele cu arhitectură Harvard mai sunt numite și „microcontrolere RISC”, RISC însemnând Reduced Instruction Set Computer, iar cele cu arhitectură von-Neuman sunt numite „microcontrolere CISC” (Complex Instruction Set Computer).



Bloc de arhitecturi Harvard vs. von Neumann

Fig 8.7 Arhitectura Harvard vs. Von-Neuman

Robotică - Îndrumar de lucrări practice

PIC16F84 are un set de 35 de instrucțiuni, în timp ce microcontrolerele INTEL sau MOTOROLA au peste 100 de instrucțiuni.

MEMORIA PROGRAM (FLASH) este utilă pentru memorarea unui program scris. Pentru că memoria ce este făcută în tehnologia FLASH poate fi programată și ștearsă mai mult decât odată, aceasta face microcontrolerul potrivit pentru aplicația noastră (activitate educațională).

EEPROM (memorie de date ce trebuie salvate atunci când microcontrolerul nu mai este alimentat). Este în mod uzual folosită pentru memorarea de date importante ce nu trebuie pierdute dacă sursa de alimentare se întrerupe neprevăzut. De exemplu, o astfel de dată este o temperatură prestabilită în regulatoarele de temperatură. Dacă în timpul întreruperii alimentării aceasta dată se pierde, va trebui să mai facem ajustarea încă o dată, la realimentare.

RAM-ul este memoria de date folosită de un program în timpul executării sale. În RAM sunt memorate toate rezultatele intermediare sau datele temporare ce nu sunt cruciale la întreruperea sursei de alimentare.

PORTUL A și PORTUL B sunt conexiuni fizice dintre microcontroler și lumea exterioară. Portul A are 5 pini, iar portul B 8 pini.

TIMER-ul LIBER (FREE RUN) este un registru de 8 biți în interiorul microcontrolerului, ce lucrează independent de program. La fiecare al patrulea impuls de ceas al oscilatorului, acesta își incrementează valoarea, până atinge maximul (255), apoi începe să numere iar de la zero.

UNITATEA DE PROCESARE CENTRALĂ are rolul unui element de conectivitate între celelalte blocuri ale microcontroller-ului. Coordonează lucrul altor blocuri și execută programul utilizatorului.

PIC16F84 are un număr total de 18 pini. Cel mai adesea, acesta se găsește într-o capsulă de tip DIP18, dar se poate găsi și într-o capsulă SMD, care este mai mică decât cea DIP. DIP este prescurtarea de la Dual In Package. SMD este prescurtarea de la Surface Mount Devices, sugerând că nu mai este necesară găurirea plăcii de montaj, pentru lipirea acestui tip de componentă.

Pinii microcontroler-ului PIC16F84 au următoarea semnificație:

Pin nr.1 **RA2**, al doilea pin la portul A. Nu are funcție adițională;

Pin nr.2 **RA3**, al treilea pin la portul A. Nu are funcție adițională;

Pin nr.3 **RA4**, al patrulea pin la portul A. TOCK1, care funcționează ca timer, se găsește de asemenea la acest pin.; Pin nr.4 MCLR, resetează intrarea și tensiunea de programare V_{pp} a microcontrolerului;

Pin nr.5 **VSS**, masă alimentare;

Pin nr.6 **RB0**, pin de zero la portul B. Intrarea Intrerupere este o funcție adiționala;

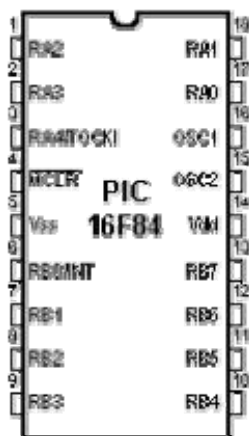


Fig. 8.8 Microcontroler PIC16F84

Pin nr.7 **RB1**, primul pin la portul B. Nu are funcție adițională;

Pin nr.8 **RB2**, al doilea pin la portul B. Nu are funcție adițională;

Pin nr.9 **RB3**, al treilea pin la portul B. Nu are funcție adițională;

Pin nr.10 **RB4**, al patrulea pin la portul B. Nu are funcție adițională;

Pin nr.11 **RB5**, al cincilea pin la portul B. Nu are funcție adițională;

Pin nr.12 **RB6**, al șaselea pin la portul B. Linia de 'Clock' în mod programare;

Pin nr.13 **RB7**, al șaptelea pin la portul B. Linia 'Data' in mod programare;

Pin nr.14 **Vdd**, polul pozitiv al sursei;

Pin nr.15 **OSC2**, pin desemnat pentru conectarea la un oscilator;

Pin nr.16 **OSC1**, pin desemnat pentru conectarea la un oscilator;

Pin nr.17 **RA2**, al doilea pin la portul A. Nu are funcție adițională;

Pin nr.18 **RA1**, primul pin la portul A. Nu are funcție adițională.

Pin nr.8 **RB2**, al doilea pin la portul B. Nu are funcție adițională;

Memoria EEPROM face mai ușoară utilizarea microcontrolerului în montaje unde se cere memorarea permanentă a diferitor parametri (coduri pentru transmițătoare, viteza motorului, frecvențele receptorului, etc.). Costul scăzut, consumul energetic redus, utilizarea ușoară și flexibilitatea sa, fac PIC16F84 aplicabil chiar și în domenii unde microcontrolerile nu au fost prevăzute înainte.

8.3.3 Schema electronică de comandă

Schema electronică de bază pentru comanda unui servomecanism este prezentată în Fig. 8.9.

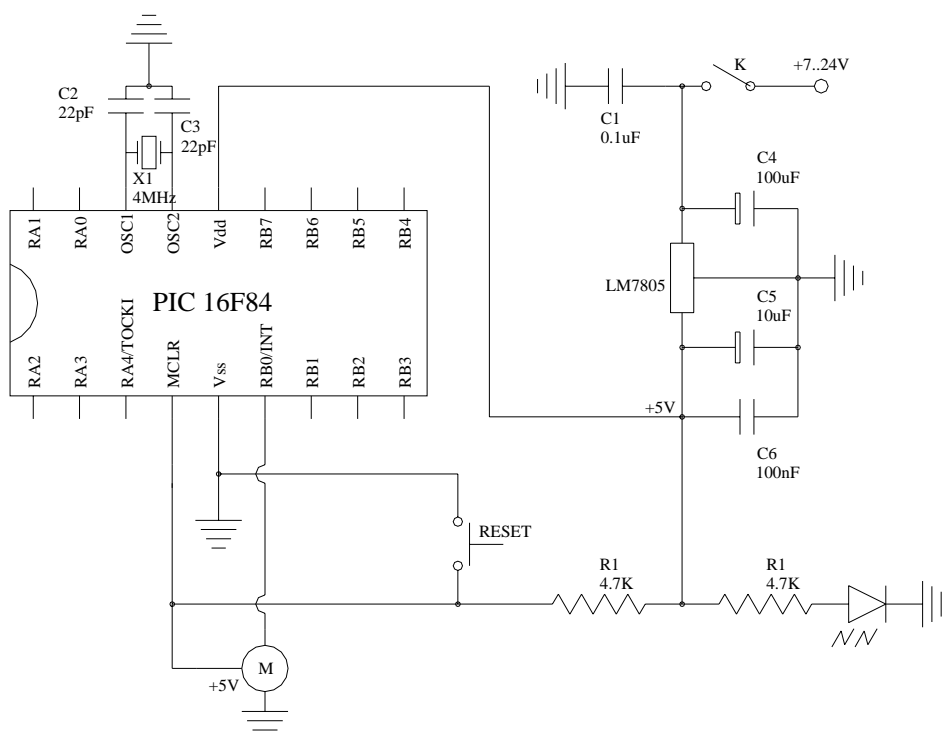


Fig. 8.9 Schema electronică de bază pentru comanda unui servomecanism

8.3.4 Exemplu de program, pentru compilator C2C

```
//servo.c
//-----
//PBO – pin la care se conecteaza SERVO
//
# pragma CLOCK_FREQ 4000000 //declararea frecventei
oscilatorului
//-----
//-----
void stanga(int cicle) //definirea unei functii pentru pozitia
limita stanga
{
char a;
a=0x01; //declararea (deschiderea) pinului de comandă,
în HEX
while(cicle)
{
PORTB=a;
delay_ms(1); //latimea impulsului pentru pozitia limita stanga
PORTB=0x00; //închiderea tuturor porturilor
delay_ms(18); //timpul dintre două impulsuri
cicle--;
}
}
//-----
//-----
void dreapta(int cicle) //definirea unei functii pentru
pozitia limita dreapta
{
char a;
a=0x01; //declararea (deschiderea) pinului de comandă,
în HEX
while(cicle)
{
PORTB=a;
delay_ms(2); //latimea impulsului pentru pozitia limita stanga
PORTB=0x00; //închiderea tuturor porturilor
delay_ms(18); //timpul dintre două impulsuri
cicle--;
}
}
//-----
//-----
```

```
main(void)
{
    set_bit( STATUS, RP0 );
    TRISB=0x00;
    clear_bit(STATUS, RP0 );
    while(1)
    {
        stanga(100);          //apelarea functiei stanga (100 cicluri –
        impulsuri)
        delay_s(5);           //pauza, 5 secunde
        dreapta(100);         //apelarea functiei dreapta (100 cicluri –
        impulsuri)
        delay_s(5);           //pauza, 5 secunde
    }
}
```

8.4 Modul de lucru

- Studenții vor primi un set de componente electronice și o placă de test;
- Aceștia vor trebui să realizeze montajul electronic de comandă, conform schemei din Fig. 1.8;
- De asemenea, vor scrie un program pentru comanda servomecanismului, care să permită acestuia să treacă prin pozițiile unghiulare de 30° , 90° și 120° , cu pauze de: 2 [s] între pozițiile 30° - 90° ; 5 [s] între pozițiile 90° - 120° și 3 [s] între pozițiile 120° - 30° .

LUCRAREA 9

COMANDA UNUI MECANISM CU DOUĂ GRADE DE MOBILITATE (PICIOR DE ROBOT PĂȘITOR) CU MICROCONTROLLER PIC16F84

9.1 Scopul lucrării

Scopul acestei lucrări este de a comanda și controla mecanismul unui picior de robot pășitor, acționat cu două servomecanisme MPX Tiny-S, utilizând un microcontroller PIC 16F84.

9.2 Servomotor MPX Tiny-S

9.2.1 Date tehnice

MPX Tiny-S este un servomecanism care oferă foarte bune performanțe în raport cu masa sa. Datele tehnice ale acestuia sunt prezentate în Tabelul 9.1.

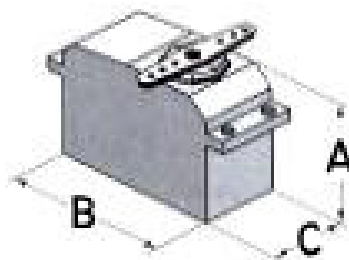


Fig. 9.1 Servomecanism MPX Tiny-S

Tabelul 9.1 Date tehnice

Dimensiune BxAxC (mm)		30x30x12
Greutate (g)		17
Moment de torsiune (Ncm)	4.8V	25

	6V	29
Viteza (60 grade)	4.8V	0.08
	6V	0.06

9.2.2 Principiu de efectuare a controlului.

Servomecanismul este ideal pentru acționarea unui element aflat în mișcare de oscilație, cu poziții limită cuprinse 0° și 180° , trecând printr-o poziție neutră la 90° . Limitarea deplasării se face mecanic, printr-un limitator atașat axului de ieșire al servomecanismului.

Comanda poziției arborelui de ieșire al servomecanismului se face prin impulsuri, cu o lățime ce variază între 1.25 și 1.75 [ms]. Acest tip de control se numește PCM (Pulse Cod Modulation). Lățimea impulsului va determina poziția unghiulară a axului.

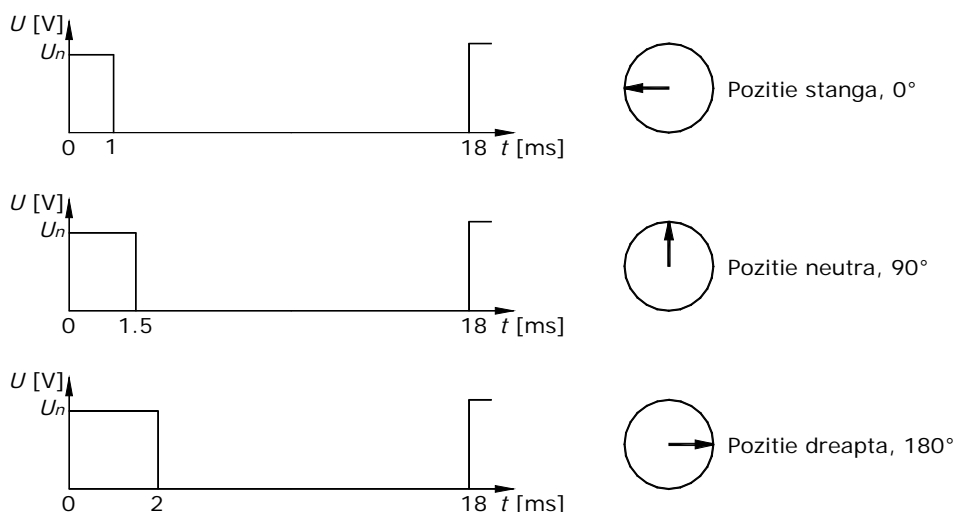


Fig. 9.2 Poziționarea axului funcție de lățimea impulsului

Cu alte cuvinte, dacă poziția neutră (90°) este dată, teoretic, de un impuls cu lățimea de 1.5 [ms], atunci când acesta va fi mai mare de 1.5 [ms] axul se va roti către poziția de 180° , iar dacă va fi

mai mic de această valoare, atunci axul se va roti către poziția de 0° (Fig. 9.2). Poziția unghiulară este direct proporțională cu lățimii impulsului, ceea ce înseamnă că, pentru a avea o poziționare la un unghi de 45° , lățimea impulsului trebuie să fie, teoretic, de 1.375 [ms]. În practică, aceste valori pot să fie puțin diferite, pentru servomecanisme diferite, datorită preciziei de montaj a componentelor

Diagrama Bloc
M51660L

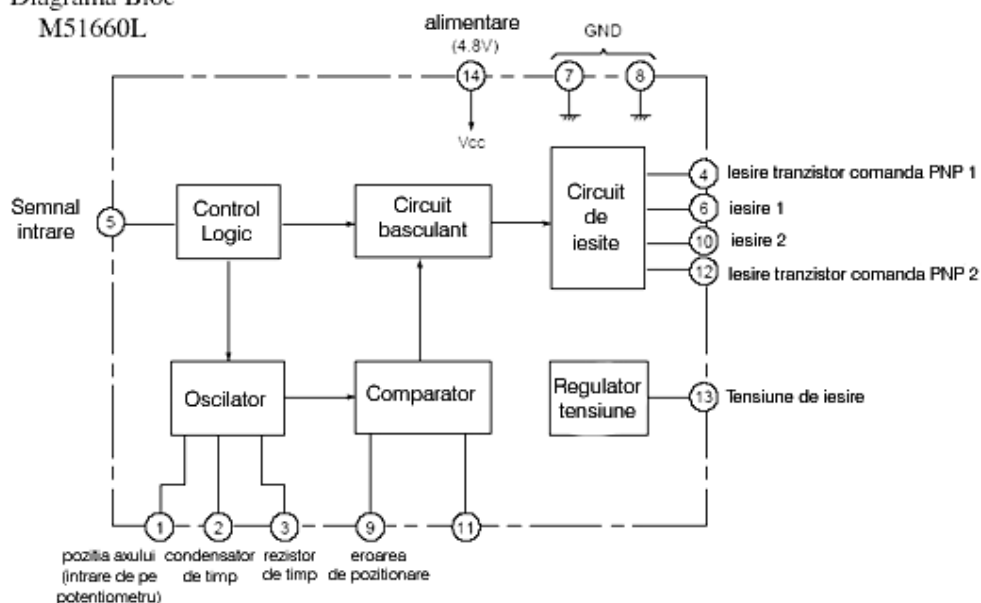


Fig. 9.3 Diagrama bloc a circuitului de comandă M51660L

mecanice, dar și a caracteristicii potențiometrului. De aceea, se recomandă o etalonare prealabilă a fiecărui servomecanism, înainte de utilizarea acestuia.

Perioada T recomandată a semnalului de comandă trebuie să fie cuprinsă între 18 și 20 [ms]. În practică, aceasta poate avea și alte valori. Drept consecință, lățimea t a impulsului va dicta poziția unghiulară, iar raportul t/T va determina viteza unghiulară a axului de ieșire.

Pentru controlul poziției acestui tip de servomecanism, firma producătoare folosește un circuit integrat specializat **M51660L**, fabricat de firma MITSUBISHI. Diagrama bloc a driver-ului (Fig. 9.3) prezintă principalele module ale acestuia și modul de comunicare dintre ele.

În cazul servomecanismului **MPX Tiny-S**, circuitul de mai sus este folosit în configurația următoare :

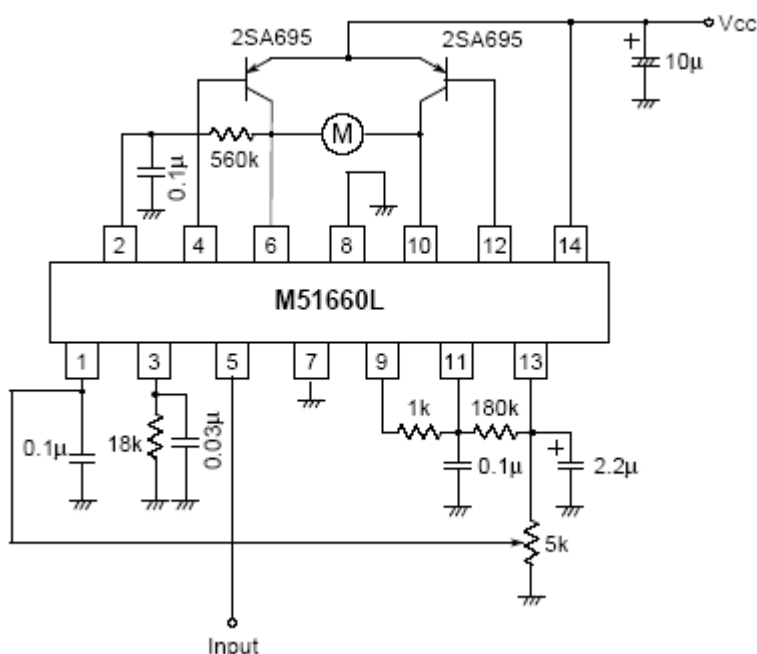


Fig. 9.4 Schema circuitului de comandă a servomecanismului MPX Tiny-S

9.3 Mecanismul piciorului

Una dintre problemele majore ale roboților mobili o constituie consumul energetic, atunci când dorim ca aceștia să fie autonomi, consum ce trebuie redus cât mai mult posibil, pentru a mări autonomia sistemului. Acumulatorii de putere mare au, în general, și o masă pe măsură, fapt ce conduce la creșterea masei generale a robotului. Acest lucru conduce la necesită motoare mai puternice, mai grele la rândul lor, consumul energetic crescând. Ținând seama de aceste probleme, s-a încercat găsirea unui mecanism pentru picior, care să diminueze consumul energetic. Astfel, structura piciorului are la bază mecanismul cu culisă de translație (vezi Fig. 9.5). Acest picior are următoarele avantaje:

- cuplu rezistent minim pe axa motorului β , ceea ce înseamnă consum de energie mai redus în faza de suport a piciorului, adică

atunci când acesta este pe sol (acest cuplu rezistent este – teoretic – zero atunci când piciorul în faza de suport este ca în poziția reprezentată în Fig. 9.5);

- distanța dintre cuplele cinematice A și D , pe direcția axei y , (distanța dintre axa elementului 1 și axa elementului 4 pe direcția menționată) este constantă și egală pentru toate picioarele, în timpul mersului, chiar dacă robotul se deplasează în teren accidentat. Aceasta înseamnă un algoritm de control mai simplu, deoarece pentru deplasarea în linie dreaptă a robotului, toate picioarele pot avea aceeași cursă unghiulară (indiferent de natura terenului).

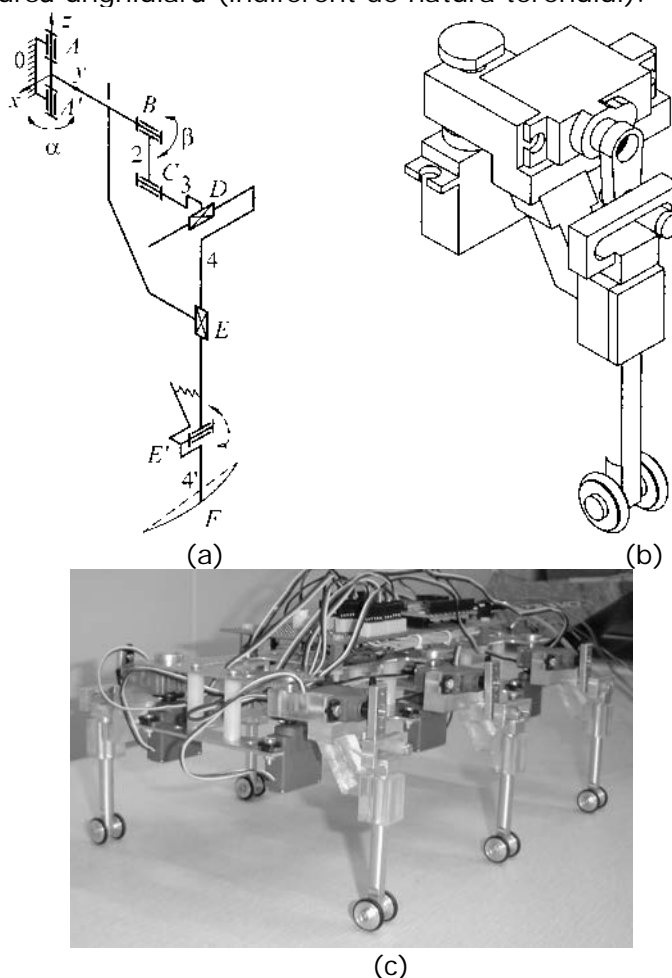


Fig. 9.5 Robot hexapod: a) structura piciorului; b) model CAD 3D al unui picior; c) imagine reală a robotului

În scopul obținerii unei traiectorii rectilinii pentru corpul robotului, în cazul deplasării în linie dreaptă înainte/înapoi, traiectoriile

extremităților picioarelor în faza de suport trebuie să fie, de asemenea, rectilinii. În cazul unui picior în buclă deschisă, având numai cuple cinematice de rotație, acest lucru înseamnă că piciorul trebuie să aibă minim trei cuple cinematice motoare:

- o cuplă pentru ridicarea/coborarea piciorului (cupla *B* în Fig. 9.5.a);
- o cuplă cinematică pentru rotația piciorului în plan orizontal (cupla *A*);
- a treia cuplă, pentru corectarea traiectoriei circulare a extremității piciorului în faza de suport (cupla cinematică *E*).

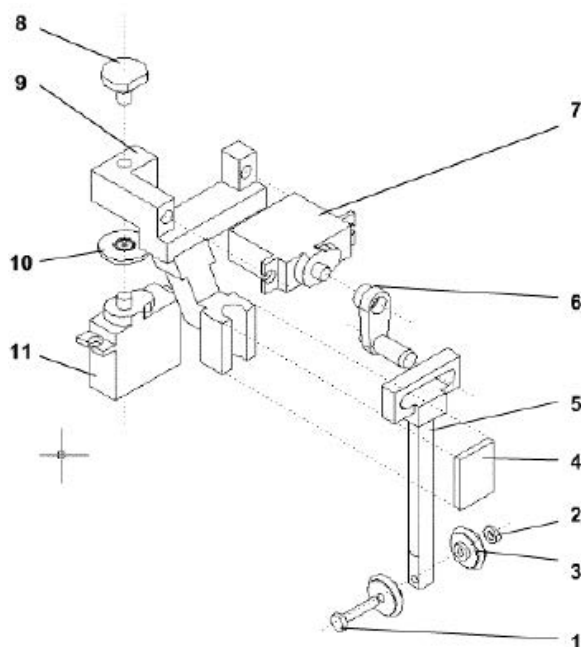


Fig 9.6 Elementele componente ale piciorului: 1-ax, 2-piuliță, 3-roată, 4- capac tijă, 5- picior propriu-zis (culisă de translație), 6-manivelă, 7-motor ridicare picior, 8-ax lagăr vertical, 9-șasiu, 10-flanșă de legătură, 11-motor rotire picior în plan orizontal

Din considerente economice (preț de cost ridicat al sistemului, utilizând trei motoare pentru fiecare picior), dar și pentru simplificarea controlului platformei, s-a utilizat o structură cu două cuple cinematice active (*A* și *B*, în Fig. 9.5.a). În asemenea condiții, traiectoria extremității piciorului în plan orizontal va fi circulară (linia continuă în figura respectivă). În scopul corectării traiectoriei, se utilizează o

cuplă cinematică pasivă, E' , „acționată” de un resort (traectoria este trasată cu linie întreruptă).

Rezolvând cinematica directă a mecanismului unui picior, obținem

$$\begin{bmatrix} x_F \\ y_F \\ z_F \end{bmatrix} = \begin{bmatrix} -\sin \alpha & -\cos \alpha \cdot \sin \beta & 0 \\ \cos \alpha & -\sin \alpha \cdot \sin \beta & 0 \\ 0 & -\cos \beta & -1 \end{bmatrix} \cdot \begin{bmatrix} l_{AB} + l_{CD} \\ l_{BC} \\ l_{DF} \end{bmatrix}, \quad (9.1)$$

ecuații ce reprezintă coordonatele extremității piciorului, în raport cu originea sistemului de axe atașat cuplei de rotație dintre mecanismul piciorului și corpul robotului.

Elementele componente ale mecanismului piciorului sunt reprezentate în Fig. 9.6.

9.4 Platforma de dezvoltare cu microcontroler PIC16F84

9.4.1 Schema electronică de comandă

Schema electronică de bază pentru comanda mecanismului piciorului, acționată de două servomecanisme este prezentată în Fig. 9.7.

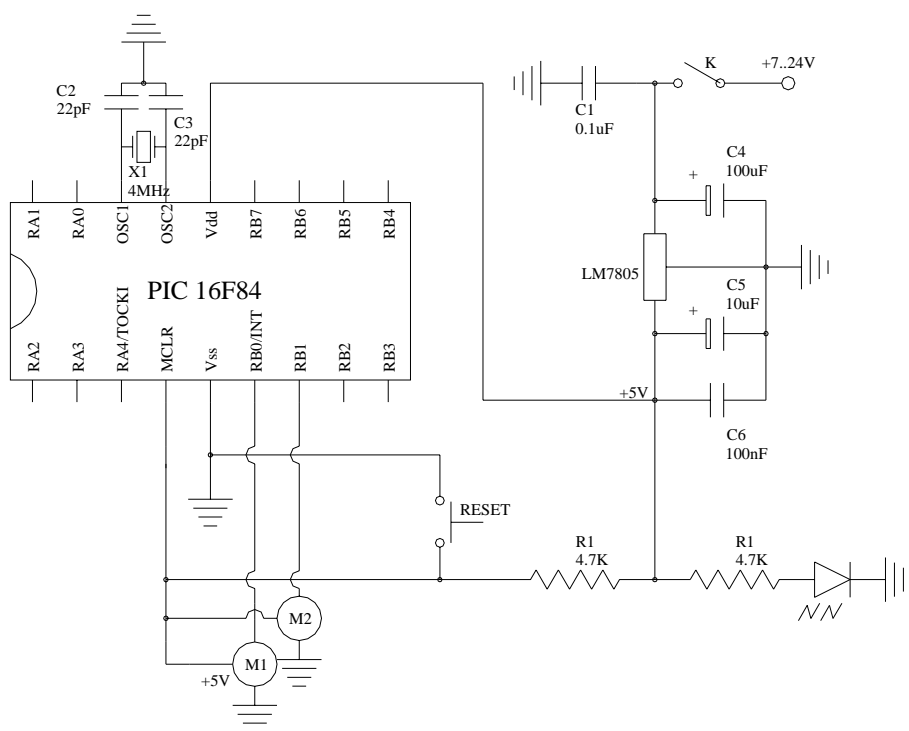


Fig. 9.7 Schema electronică de bază pentru comanda piciorului

9.4.2 Exemplu de program, pentru compilator C2C

```
// PICIOR.c
//
// # pragma CLOCK_FREQ 4000000
//
//
//char a;
//a=0x01; // PB0 - MOTOR 2_V PORT
//char b;
//b=0x02; // PB1 - MOTOR 2_H PORT
//
```

Robotică - Îndrumar de lucrări practice

```
//-----  
//-----  
void f1(int cicle)  
{  
while(cicle)  
{  
PORTB=0x01;  
delay_ms(1);          // MOTOR 2_V JOS, DELAY FOR "1" SIGNAL  
delay_us(250);  
delay_us(250);  
PORTB=0x02;  
delay_ms(1);          // MOTOR 2_H SPATE, DELAY FOR "1" SIGNAL  
delay_us(250);  
PORTB=0x00;  
delay_ms(40);         // DELAY FOR "0" SIGNAL  
cicle--;  
}  
}  
//-----  
//-----  
void f2(int cicle)    //  
{  
while(cicle)  
{  
PORTB=0x01;  
delay_us(250);         // MOTOR 2_V SUS, DELAY FOR "1"  
SIGNAL  
delay_us(250);  
PORTB=0x02;  
delay_ms(1);          // MOTOR 2_H SPATE, DELAY FOR "1" SIGNAL  
delay_us(250);  
PORTB=0x00;  
delay_ms(40);         // DELAY FOR "0" SIGNAL  
delay_us(250);  
delay_us(250);  
delay_us(250);  
delay_us(150);  
cicle--;  
}  
}  
//-----  
//-----  
main(void)  
{
```

```
set_bit(STATUS, RP0);
TRISB=0x00;
clear_bit(STATUS, RP0);
//
//
while(1)
{
    f1(20);                // JOS_SPATE
    f2(20);                // SUS_SPATE
}
```

9.5 Modul de lucru

- Studenții vor primi un set de componente electronice și o placă de test;
- Aceștia vor trebui să realizeze montajul electronic de comandă, conform schemei din Fig. 9.7;
- De asemenea, vor scrie un program pentru comanda piciorului, care să simuleze mișcarea acestuia în cazul mersului înainte al robotului.

LUCRAREA 10

CONTROLUL MIȘCĂRII ROBOTULUI OMNIDIRECȚIONAL CU ROȚI UNIVERSALE - ROBOTINO

10.1 Roata universală

Roata universală (Fig. 10.1) este un exemplu de roată specială, ce are un număr de role pasive montate pe circumferința acesteia. Arborii rolor au axele perpendiculare pe axa principală de rotație a roții. Roata este acționată ca și o una convențională, în timp ce rolele îi permit o deplasare liberă pe o direcție perpendiculară pe direcția

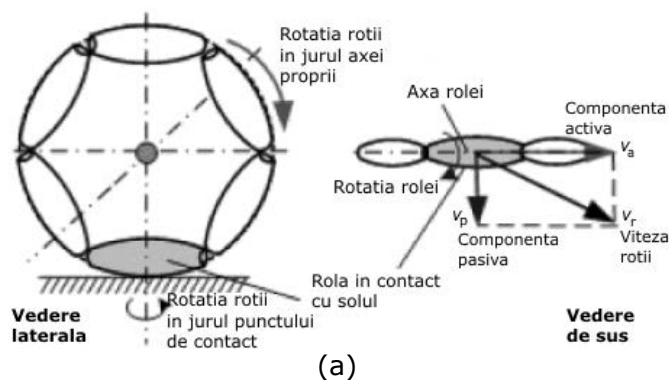


Fig. 10.1 Roata universală: a) grade de libertate; b) soluție constructivă

normală de tracțiune, astfel că, în final, roata universală are trei grade de libertate: rotația în jurul axei proprii, rotația în jurul axei verticale ce trece prin punctul de contact cu solul și rotația liberă a roților în jurul axelor proprii (Fig. 10.1.a). Trei asemenea roți universale vor oferi o deplasare omnidirecțională, platforma mobilă fiind static determinată (Fig. 10.2). Pot fi utilizate mai multe asemenea roți, pasive sau active, pentru a suporta încărcări mai mari sau pentru a oferi o stabilitate mai bună însă, în aceste situații este necesar un sistem de suspensie, iar masa totală a vehiculului crește.

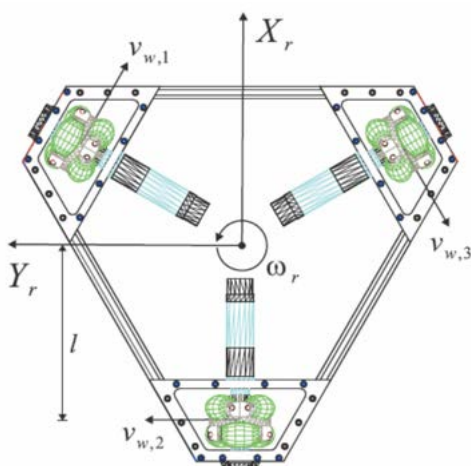


Fig. 10.2 Vehicul cu trei roți universale

Roțile universale sunt compacte, se găsesc în comerț, sunt ușor de montat și prezintă bune proprietăți omnidirecționale. Datorită roților pasive de pe circumferința acestora, roțile universale sunt sensibile la neregularitățile terenului pe care se deplasează, pot suporta doar încărcări de până la 100 kg și prezintă discontinuități ale contactului cu solul, ca urmare a golului dintre două role, aceste discontinuități introducând vibrații verticale în robot.

Problema discontinuităților contactului dintre roată și suprafața pe care se deplasează poate fi rezolvată prin utilizarea a două rânduri de role intercalate (Fig. 10.3). Acest lucru va introduce probleme de control și odometrie, datorită faptului că punctul de contact al roții cu solul se deplasează pe direcția axei roții, la trecerea de pe o rolă pe cealaltă. Acest aspect poate, de asemenea, introduce și vibrații orizontale în robot. Dacă distanța dintre cele două rânduri de role este mică în comparație cu raza vehiculului, problemele menționate pot fi

neglijate. Pentru eliminarea totală a acestora, se poate folosi o soluție de roată cu role alternante (Fig. 10.4).

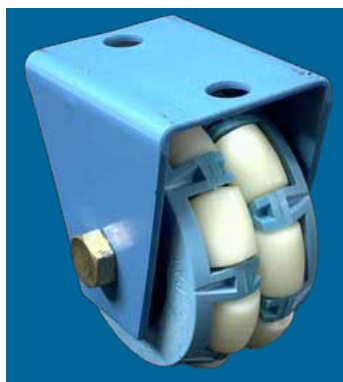


Fig. 10.3 Roata universală cu două rânduri de role

Acest tip de roată este o altă soluție utilizată pentru reducerea vibrațiilor cauzate de discontinuitățile rolelor succesive. Punctele de contact cu solul sunt în linie, minimizând vibrațiile orizontale.



Fig. 10.4 Roata universală cu role alternante

10.2 Elemente constructive ale robotului

Avantajul vehiculelor omnidirecționale este acela că se pot deplasa pe orice direcție, fără a fi nevoie să se rotească (Fig. 10.5). Elementul cheie al acționării omnidirecționale este așa numita roată omnidirecțională sau roata multidirecțională sau cu role (Fig. 10.6). Rolele, având de obicei forma de butoi, sunt atașate la corpul central

al roții, axa de rotație a acestora făcând un anumit unghi cu cea a roții.

Roata omnidirecțională poate fi acționată activ prin intermediul motorului și, de asemenea, să se deplaseze lateral, pasiv, prin intermediul rotelor din structura sa. Interacționând cu alte două roți, se pot genera mișcări pe orice direcție, care deviază de la mișcarea principală de acționare a roții.

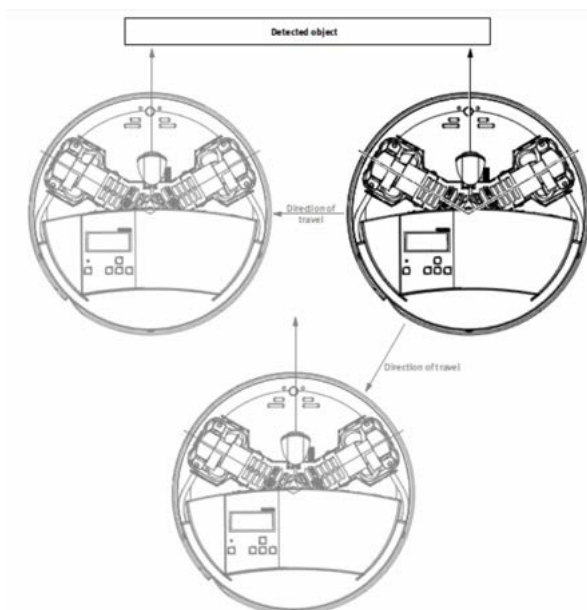


Fig. 10.5 Deplasare omnidirecțională

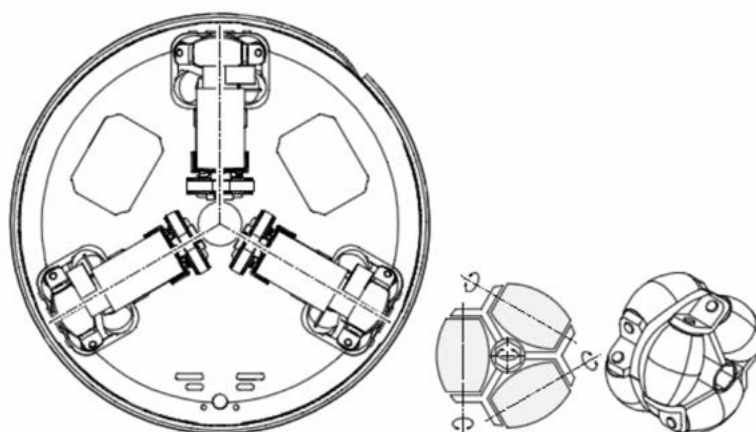


Fig. 10.6 Roată omnidirecțională (universală)

Într-o acționare simetrică, roțile sunt dispuse sub un unghi de 120° , după cum se poate vedea în Fig. 10.6 și Fig. 10.7.



Fig. 10.7 Dispunerea simetrică a roților

Avantajul acestui tip de sistem este acela că, folosind diverse viteze pentru motoare, robotul este capabil să se deplaseze pe orice direcție, fără a fi nevoie să se rotească. Acesta nu necesită să se întoarcă de la obiectul pe care îl urmărește, menținându-l, în permanență, în vizorul său. Alte avantaje ale acestuia pot fi menționate în cele ce urmează:

- un vehicul omnidirecțional poate vira în jurul axei proprii;
- masa robotului este distribuită pe toate cele trei roți, echilibrul acestuia fiind îmbunătățit;
- nu este necesar un mecanism de direcție;
- este simplu și robust din punct de vedere mecanic.

Unul dintre dezavantaje este consumul energetic relativ mare, în cazul deplasării înainte, deoarece această mișcare are loc odată cu rostogolirea unui număr mare de role și, implicit, cu frecări mari.

Pot fi folosite diferite tipuri de roți, în funcție de suprafața pe care se deplasează robotul. Roțile folosind role din plastic moale sau poliuretan se pretează la suprafețe dure și netede, cum ar fi sticla sau gresia, în timp ce utilizarea de role dure este de preferat în cazul deplasării pe suprafețe moi, cum ar fi covorul sau cartonul. O altă diferență este dată de numărul de role ce se poate utiliza, precum și numărul de rânduri de role.

Robotino poate transla pe orice direcție (înainte, înapoi, lateral, etc.) și poate vira în jurul axei proprii, având, în acest fel, trei grade de mobilitate (două translații și o rotație).

Componentele principale ale robotului sunt prezentate în Fig. 10.8, într-o vedere axonometrică explodată, precum și în Fig. 10.9, ce reprezintă o fotografie din față a vehiculului omnidirecțional ROBOTINO.

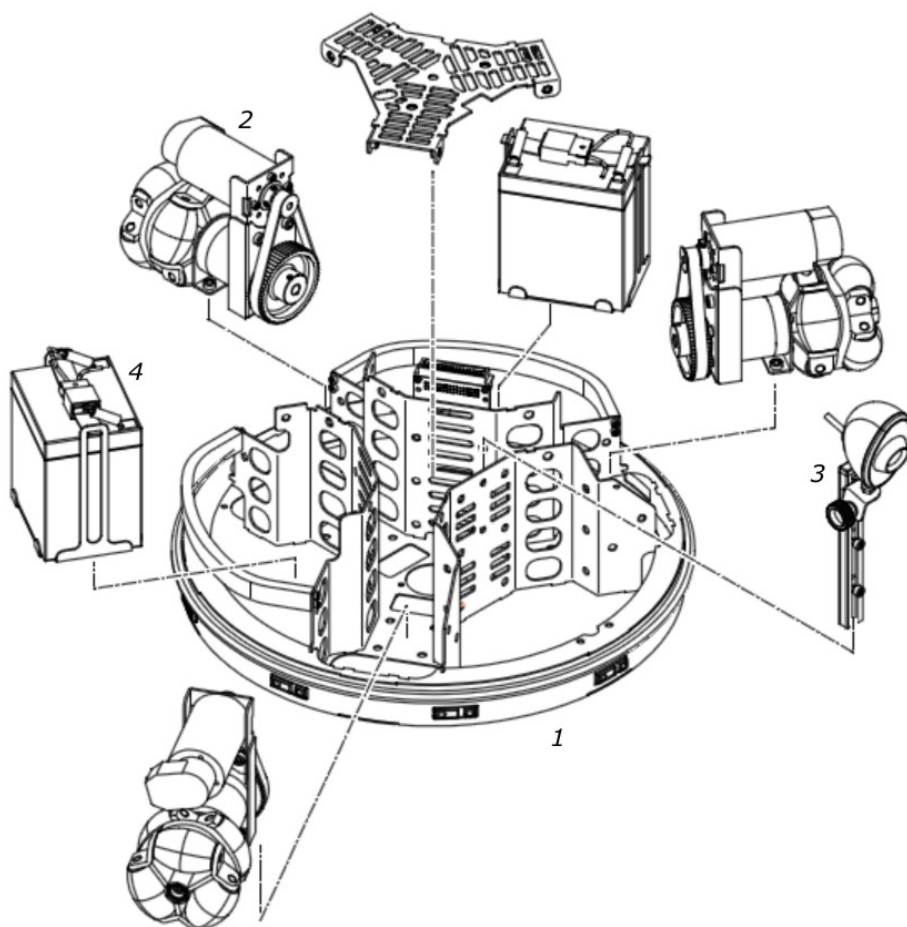


Fig. 10.8 Componentele principale ale robotului - vedere explodată: 1 - șasiu cu senzor de contact și senzori infraroșu; 2 - trei grupuri acționare (motoreductor cu encoder, transmisie prin curea dințată și roată universală dublă); 3 - o platformă de lucru cu cameră web; 4 - doi acumulatori

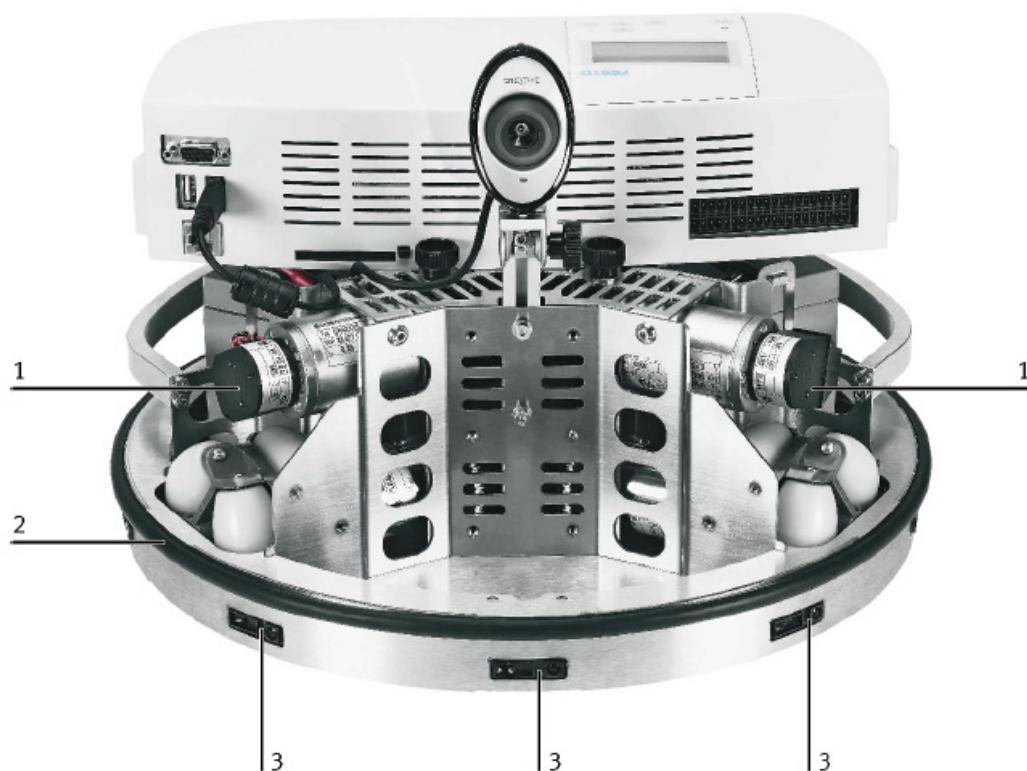


Fig. 10.9 Vehicul omnidirecțional ROBOTINO - foto din față: 1 - encodere incrementale motoare; 2 - bară de șoc cu senzor anticoliziune; 3 - senzori infraroșu de distanță

Detalii privind dispunerea senzorilor, precum și a altor componente constructive ale robotului sunt reprezentate și în Fig. 10.10. Bara de șoc este reprezentată de un tub elastic, dispus pe circumferința circulară a corpului robotului, având proprietatea că, atunci când robotul intră în coliziune cu un obstacol, deformarea acestuia închide un contact electric, oprind deplasarea vehiculului (reprezintă senzorul - soluție de avarie, în cazul în care obstacolele nu sunt detectate de ceilalți senzori).

Senzorii infraroșu de determinare a distanței dintre corpul robotului și obstacolele din jurul acestuia sunt în număr de nouă și sunt dispuși, echidistanți, pe întreaga circumferință a acestuia, sub unghiuri de 40°.

Camera video este montată pe daxa 0°, reprezentând direcția de deplasare a acestuia către direcția înainte.

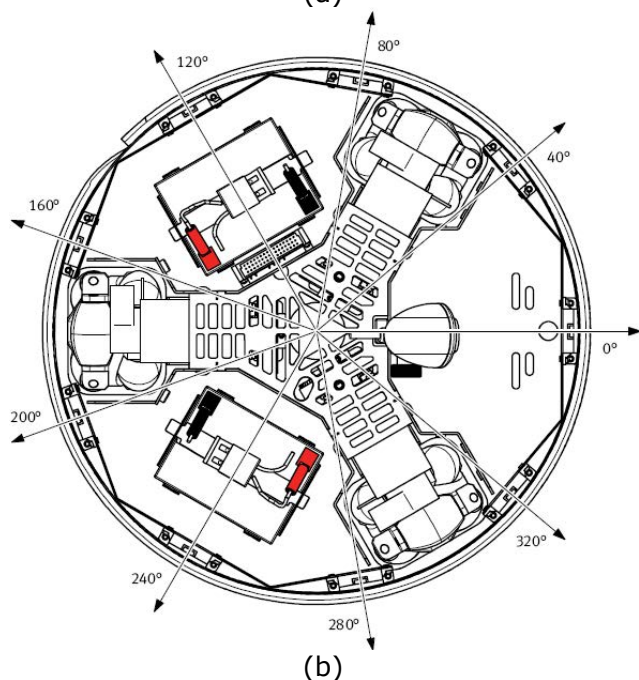
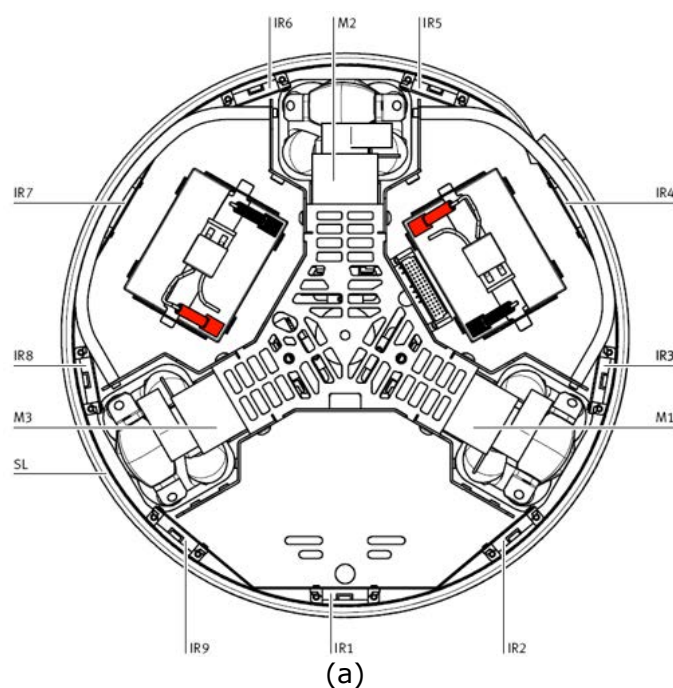


Fig. 10.10 Detalii privind dispunerea senzorilor și a altor componente

Sub corpul vehiculului se mai află doi senzori optici digitali (de difuzie), care-i permit robotului să se orienteze după un traseu marcat pe suprafața pe care acesta se deplasează. De asemenea, pe axa 0°, chiar în fața vehiculului, orientat tot spre suprafața de suport, se află un senzor optic analog de proximitate.

Grupul de acționare al unei roți universale din structura robotului ROBOTINO, este reprezentat în Fig. 10.11, unde s-au făcut notațiile: 1 - motor de c.c., 12 V, 2 - encoder incremental cu 2048 impulsuri pe rotație; 3 - role pasive (libere la rotația în jurul axelor proprii); 4 - reductor planetar cu raport de transmitere de 1:16; 5 - transmisie prin curea dințată.

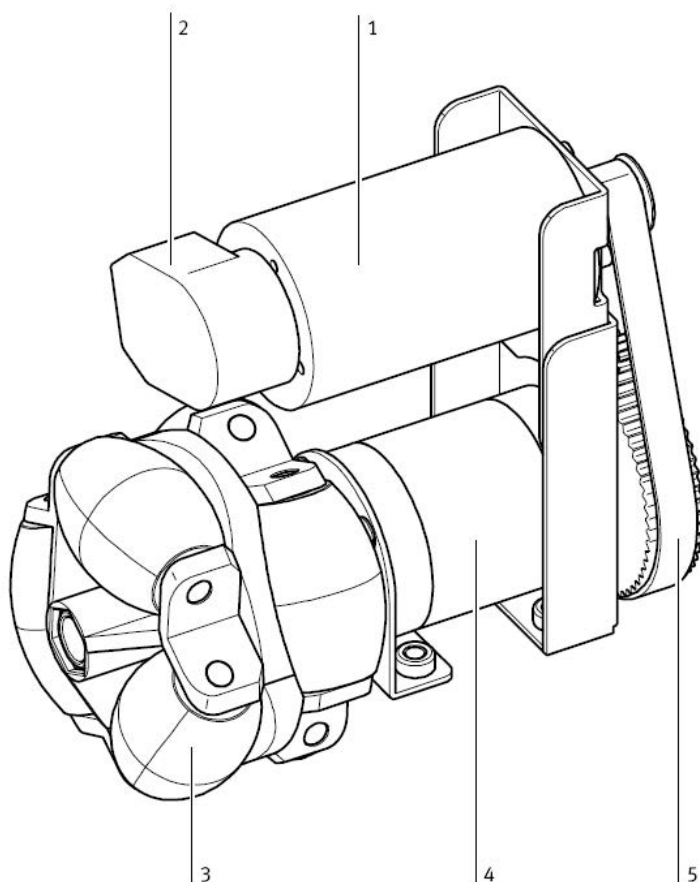


Fig. 10.11 Componentele unui grup de acționare a unei roți universale

10.3 Cinematica robotului

Aționarea unui vehicul omnidirecțional este destul de complicată. În cazul unui robot cu trei roți, sunt necesare trei motoare. Deoarece fiecare dintre aceste motoare poate primi trei comenzi de control (rotație înainte, rotație înapoi sau stop), există 27 opțiuni pentru toate cele trei motoare (Fig. 10.12). Aceste 27 combinații de comenzi permit robotului să fie controlat pentru a se deplasa pe orice direcție. Tabelul ce urmează (Tabelul 10.1) prezintă cele 27 de opțiuni posibile de comenzi, precum și direcția de mișcare rezultată. La acest nivel, nu sunt luate în considerație variațiile vitezelor celor trei motoare, viteza de rotație a acestora fiind aceeași.

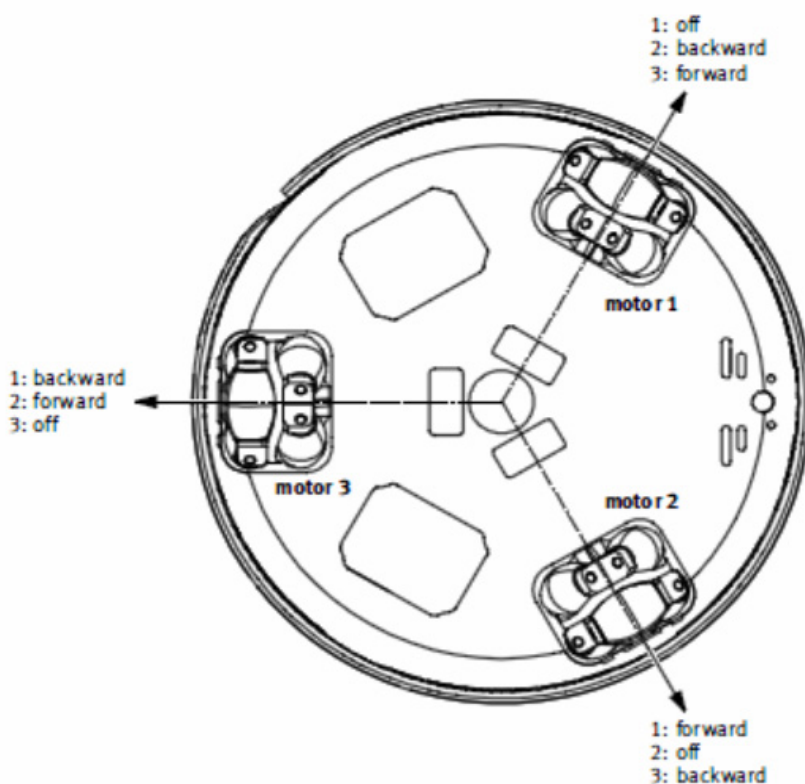


Fig. 10.12 Comenzi posibile ale motoarelor

Tabelul 10.1 Acționare motoare și direcție de deplasare robot

Motor 1	Motor 2	Motor 3	Direcția de deplasare
Rotatie în sens orar	Rotatie în sens orar	Rotatie în sens orar	Rotatie în sens orar, fara a se deplasa (viraj în jurul axei proprii)
Rotatie în sens orar	Rotatie în sens orar	Stop	Rotatie în sens orar, descriind o traiectorie cu raza mica
Rotatie în sens orar	Rotatie în sens orar	Rotatie în sens trigonometric	Rotatie în sens orar, descriind o traiectorie cu raza mare
Rotatie în sens orar	Stop	Rotatie în sens orar	Rotatie în sens orar, descriind o traiectorie cu raza mica
Rotatie în sens orar	Stop	Stop	Rotatie în sens orar, descriind o traiectorie cu raza medie
Rotatie în sens orar	Stop	Rotatie în sens trigonometric	Translatie pe directia 300°
Rotatie în sens orar	Rotatie în sens trigonometric	Rotatie în sens orar	Rotatie în sens orar, descriind o traiectorie cu raza mare
Rotatie în sens orar	Rotatie în sens trigonometric	Stop	Translatie pe directie 0°
Rotatie în sens orar	Rotatie în sens trigonometric	Rotatie în sens trigonometric	Rotatie în sens trigonometric, descriind o traiectorie cu raza mare
Stop	Rotatie în sens orar	Rotatie în sens orar	Rotatie în sens orar, descriind o traiectorie cu raza mica
Stop	Rotatie în sens orar	Stop	Rotatie în sens orar, descriind o traiectorie cu raza medie
Stop	Rotatie în sens orar	Rotatie în sens	Translatie pe directia 240°

Robotică - Îndrumar de lucrări practice

		trigonometric	
Stop	Stop	Rotatie in sens orar	Rotatie in sens orar, descriind o traiectorie cu raza medie
Stop	Stop	Stop	Stationare
Stop	Stop	Rotatie in sens trigonometric	Rotatie in sens trigonometric, descriind o traiectorie cu raza medie
Stop	Rotatie in sens trigonometric	Rotatie in sens orar	Translatie pe directia 60°
Stop	Rotatie in sens trigonometric	Stop	Rotatie in sens trigonometric, descriind o traiectorie cu raza medie
Stop	Rotatie in sens trigonometric	Rotatie in sens trigonometric	Rotatie in sens trigonometric, descriind o traiectorie cu raza mica
Rotatie in sens trigonometric	Rotatie in sens orar	Rotatie in sens orar	Rotatie in sens orar, descriind o traiectorie cu raza mare
Rotatie in sens trigonometric	Rotatie in sens orar	Stop	Translatie pe directia 180°
Rotatie in sens trigonometric	Rotatie in sens orar	Rotatie in sens trigonometric	Rotatie in sens trigonometric, descriind o traiectorie cu raza mare
Rotatie in sens trigonometric	Stop	Rotatie in sens orar	Translatie pe directia 120°
Rotatie in sens trigonometric	Stop	Stop	Rotatie in sens trigonometric, descriind o traiectorie cu raza medie
Rotatie in	Stop	Rotatie in	Rotatie in sens

sens trigonometric		sens trigonometric	trigonometric, descriind o traiectorie cu raza mica
Rotatie in sens trigonometric	Rotatie in sens trigonometric	Rotatie in sens orar	Rotatie in sens trigonometric, descriind o traiectorie cu raza mare
Rotatie in sens trigonometric	Rotatie in sens trigonometric	Stop	Rotatie in sens trigonometric, descriind o traiectorie cu raza mica
Rotatie in sens trigonometric	Rotatie in sens trigonometric	Rotatie in sens trigonometric	Rotatie in sens trigonometric, fara a se deplasa (viraj in jurul axei proprii)

Pentru modelarea cinematică a roboților mobili cu roți, sunt făcute câteva ipoteze. Roboții sunt considerați ca fiind un corp rigid, ce se deplasează pe un plan orizontal, iar roțile, aflate în plan vertical, au câte un singur punct de contact cu solul. Mai mult, se presupune că nu există alunecări, atunci când roțile accelerează sau frânează, iar orientarea acestora în jurul unei axe verticale poate avea loc fără frecare.

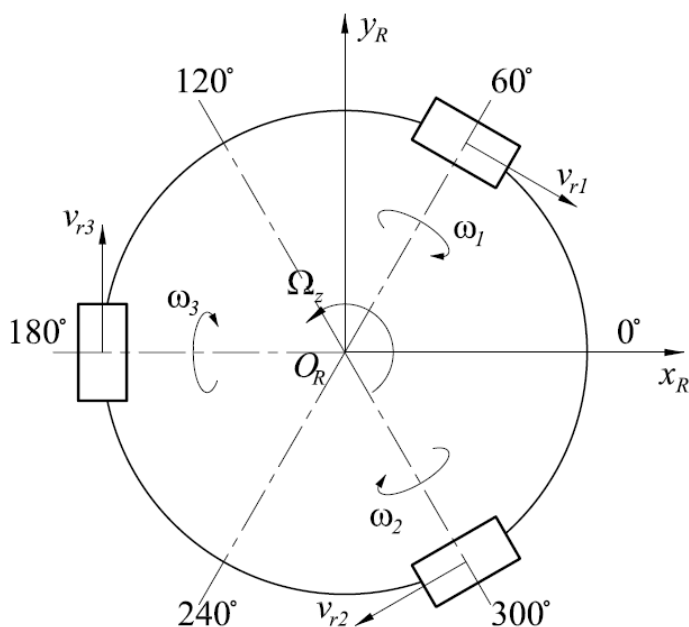


Fig. 10.13 Parametri cinematici

10.3.1 Cinematica directă

$$\begin{bmatrix} v_{Rx} \\ v_{Ry} \\ \Omega_z \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} & 0 \\ -\frac{1}{3} & -\frac{1}{3} & \frac{2}{3} \\ -\frac{1}{3 \cdot l} & -\frac{1}{3 \cdot l} & -\frac{1}{3 \cdot l} \end{bmatrix} \cdot \begin{bmatrix} v_{r1} \\ v_{r2} \\ v_{r3} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} \cdot r & -\frac{1}{\sqrt{3}} \cdot r & 0 \\ -\frac{1}{3} \cdot r & -\frac{1}{3} \cdot r & \frac{2}{3} \cdot r \\ -\frac{1}{3 \cdot l} \cdot r & -\frac{1}{3 \cdot l} \cdot r & -\frac{1}{3 \cdot l} \cdot r \end{bmatrix} \cdot \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \quad (10.1)$$

$$\begin{cases} v_{Rx} = \frac{1}{\sqrt{3}} \cdot r \cdot (\omega_1 - \omega_2) \\ v_{Ry} = \frac{1}{3} \cdot r \cdot (-\omega_1 - \omega_2 + 2 \cdot \omega_3) \\ \Omega_z = \frac{1}{3 \cdot l} \cdot r \cdot (-\omega_1 - \omega_2 - \omega_3) \end{cases} \quad (10.2)$$

10.3.2 Cinematica inversă

$$\begin{bmatrix} v_{R1} \\ v_{R2} \\ v_{R3} \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{3}}{2} & -l & -\frac{1}{2} \\ -0 & -l & 1 \\ -\frac{\sqrt{3}}{2} & -l & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} v_{Rx} \\ v_{Ry} \\ \Omega_z \end{bmatrix} \quad (10.3)$$

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \cdot \begin{bmatrix} \frac{\sqrt{3}}{2} & -l & -\frac{1}{2} \\ -0 & -l & 1 \\ -\frac{\sqrt{3}}{2} & -l & -\frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} v_{Rx} \\ v_{Ry} \\ \Omega_z \end{bmatrix} \quad (10.4)$$

$$\begin{cases} \omega_1 = \frac{1}{r} \cdot \left(\frac{\sqrt{3}}{2} \cdot v_{Rx} - l \cdot v_{Ry} - \frac{1}{2} \cdot \Omega_z \right) \\ \omega_2 = \frac{1}{r} \cdot (l \cdot v_{Ry} + \Omega_z) \\ \omega_3 = \frac{1}{r} \cdot \left(-\frac{\sqrt{3}}{2} \cdot v_{Rx} - l \cdot v_{Ry} - \frac{1}{2} \cdot \Omega_z \right) \end{cases} \quad (10.5)$$

10.3.3 Alte elemente de calcul

c_r - circumferinta rotii (distanța parcursă de o roată, la o rotație completă a acesteia)

d_r - diametrul rotii

$$c_r = \pi \cdot d_r \quad (10.6)$$

$$d_r = 80 \text{ [mm]} \quad (10.7)$$

$$c_r = \pi \cdot 80 = 251.327 \text{ [mm]} \quad (10.8)$$

n_r - numărul de rotații complete ale rotii

l_r - distanța parcursă de o roată, la un număr n_r de rotații complete

$$l_r = n_r \cdot c_r = n_r \cdot \pi \cdot d_r \quad (10.9)$$

L_R - distanța parcursă de robot

$$L_R = \frac{l_r}{\sin 60^\circ} \quad (10.10)$$

$$n_r = \frac{l_r}{c_r} = \frac{L_R \cdot \sin 60^\circ}{\pi \cdot d_r} \quad (10.11)$$

i - raport de transmitere motoreductor + transmisie prin curea dintată

$$i = \frac{1}{16} \quad (10.12)$$

n_M - numărul de rotații complete ale arborelui motorului, pentru o distanță, L_R , parcursă de robot

$$n_M = n_r \cdot 16 = \frac{16 \cdot L_R \cdot \sin 60^\circ}{\pi \cdot d_r} \quad (10.13)$$

Robotică - Îndrumar de lucrări practice

2048 - numărul de incrementi la encoderul motorului, pentru o rotație completă a arborelui acestuia

n_i - numărul de incrementi la encoderul motorului, pentru o distanță, L_R , parcursă de robot

$$n_i = n_M \cdot 2048 = \frac{2048 \cdot 16 \cdot L_R \cdot \sin 60^\circ}{\pi \cdot d_r} \quad (10.14)$$

10.4 Modul de lucru

- Studenții vor fi familiarizați cu modul de funcționare al robotului, prezentându-le și componentele principale ale acestuia;
- Aceștia vor trebui să întocmească unele rutine de deplasare ale robotului pe direcțiile de dispunere ale roților (0° , 60° , 120° , 180° , 240° , 300°);
- De asemenea, pe baza acestor rutine simple, vor întocmi o rutină care să permită deplasarea robotului după o traiectorie contur închis, prin translații elementare, fără orientarea acestuia în jurul axei verticale.