

Etapa II: Prototipul 1 (Autonomous Wheelchair 2DW/2FW): Proiectarea conducerii si navigatiei bazate pe sistem video-biometric si sistem laser a scaunului cu rotile;

Prototipul 2 (Autonomous Robot 2DW/1FW, 2DW/2FW): Proiectarea conducerii si integrarii in linii de fabricatie flexibila, in regim de laborator, a sistemelor robotice autonome 2DW/1FW si 2DW/2FW echipate cu manipuloare 6-DOF;

Prototipul 3 (Autonomous Omnidirectional Vehicle 4DW/SW): Proiectarea conducerii de la distanta si a navigatiei vehiculului autonom 4DW/SW echipat cu manipulator 6-DOF, pe diferite tipuri de carosabil.

Rezumat: Prototipul 1: Scaun cu rotile pentru persoane varstnice si cu dizabilitati severe, echipat cu structura de conducere si navigatie bazat pe biometria fetei, a ochiului si sistem laser. Scaunul are un structura de conducere sliding-mode si un sistem de navigatie si evitare obstacole bazat pe miscarea pupilei ochiului si laser, putand fi utilizat si de persoane cu dizabilitati severe. **Prototipul 2:** Modelarea, utilizand Retele Petri Hibrice(RPH), a liniilor mecatronice de asamblare/dezasamblare (Hara&Horstman)si deprelucrare/reprelucrare (Festo) deservite de roboti autonomiechipati cu manipuloar. **Prototipul 3:** proiectarea conducerii de la distanta si navigatia vehiculului autonom 4DW/SW echipat cu manipulator 6-dof, in interior si exterior, pe diferite tipuri de carosabil pentru deservirea proceselor de productie din industria sticlei

Activitatea II.1: Proiectarea structurii de conducere sliding-mode si backstepping, pentru urmarirea unei traiectorii impuse, bazata pe modelul cinematic al Prototipului 1.

Arhitectura conducerii sliding-mode cu timp continuu a WMR cu 2 DW/2FW este prezentată în Fig. 1. Această arhitectură de conducere permite robotului mobil (scaun cu rotile pentru persoane cu dizabilitati) să urmărească o traiectorie dorită cu un profil de viteză impus. Trajectory-tracking consideră că robotul urmărește un WMR virtual care se deplasează pe traiectoria dorită cu profilul de viteză impus (a se vedea Fig. 2).

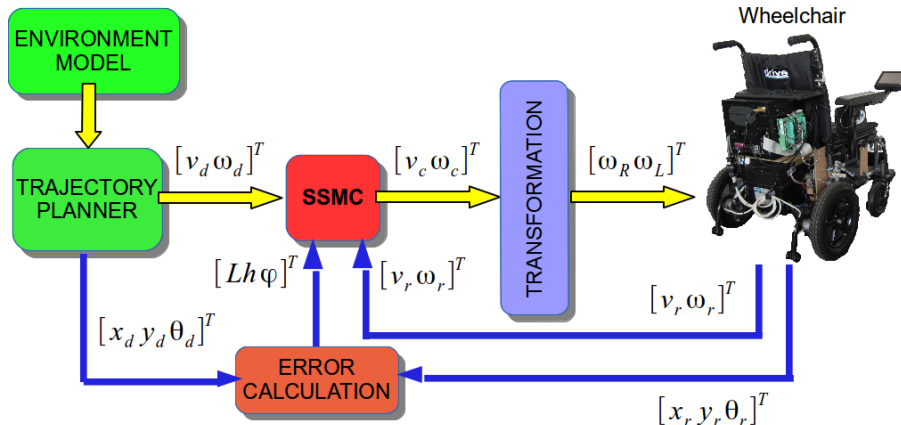


Figura 1. Arhitectura conducerii sliding-mode cu timp continuu a robotului mobil cu două roți motoare.

Modelul cinematic al robotului este:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \cos(\theta_r) & 0 \\ \sin(\theta_r) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_r \\ \omega_r \end{bmatrix}$$

unde x_r și y_r reprezintă coordonatele Carteziene ale centrului geometric, v_r este viteza liniară a robotului mobil, θ_r reprezintă orientarea și ω_r reprezintă viteza unghiulară a robotului în jurul centrului de greutate.

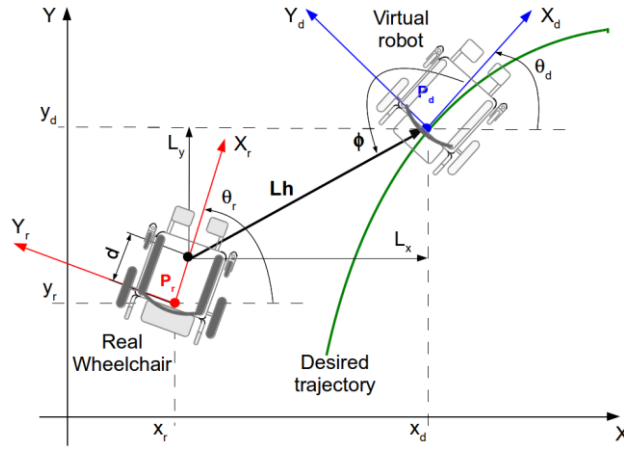


Figura 2. Definierea erorilor de urmărire (L_h , ϕ și θ_e) în cazul urmăririi unei traiectorii predefinite

Când robotul mobil încercă să urmărească traiectoria impusă se obțin următoarele erori de urmărire (Fig. 2). Erorile de urmărire sunt date de relația:

$$\begin{aligned} L_x &= x_d - x_r - d \cdot \cos(\theta_r) \\ L_y &= y_d - y_r - d \cdot \sin(\theta_r) \\ \theta_e &= \theta_d - \theta_r \end{aligned}$$

Derivatele erorilor de urmărire sunt:

$$\begin{aligned} \dot{L}_x &= v_d \cdot \cos(\theta_d) - v_r \cdot \cos(\theta_r) + d \cdot \omega_r \cdot \sin(\theta_r) \\ \dot{L}_y &= v_d \cdot \sin(\theta_d) - v_r \cdot \sin(\theta_r) - d \cdot \omega_r \cdot \cos(\theta_r) \end{aligned}$$

de unde rezultă:

$$\dot{L}_h = -v_d \cdot \cos(\phi) + v_r \cdot \cos(\gamma) + d \cdot \omega_r \cdot \sin(\gamma)$$

$$\begin{aligned} \dot{\phi} &= \frac{1}{L_h} \cdot (v_d \cdot \sin(\phi) - v_r \cdot \sin(\gamma)) + \\ &+ d \cdot \omega_r \cdot \cos(\gamma) - L_h \cdot \omega_d \end{aligned}$$

unde: $\gamma = \theta_d - \theta_r + \phi$

Dinamica erorilor de urmărire este:

$$\dot{e} = A(e, t) + B(e, t) \cdot u$$

unde $u = [v_r, \omega_r]^T$ este vectorul intrărilor, iar $e = [L_{h_e}, \psi_e]^T$, iar

$$A(e, t) = \begin{bmatrix} -v_d \cdot \cos(\phi_e + \phi_d) \\ \frac{v_d \cdot \sin(\phi_e + \phi_d) - (L_{h_e} + L_{h_d}) \cdot \omega_d}{L_{h_e} + L_{h_d}} \end{bmatrix}$$

$$B(e, t) = \begin{bmatrix} \cos(\gamma_e) & d \cdot \sin(\gamma_e) \\ \frac{-\sin(\gamma_e)}{L_{h_e} + L_{h_d}} & \frac{d \cdot \cos(\gamma_e)}{L_{h_e} + L_{h_d}} \end{bmatrix}$$

În continuare se pot defini suprafețele de comutație (sliding surfaces):

$$s(t) = e(t) + k_e \cdot \int_0^t e(\tau) \cdot d\tau$$

unde $k_e > 0$.

Legea de conducere utilizată este de forma:

$$u_c(t) = \frac{u_1(t) - k_e \cdot e(t) - A(e, t)}{B(e, t)}$$

unde:

$$u_1(t) = -k_1 \cdot \text{sign}(s(t)) \cdot |s(t)|^{1/2} - k_2 \cdot s(t) + \dot{\sigma}(t)$$

$$\dot{\sigma}(t) = -k_3 \cdot \text{sign}(s(t)) - k_4 \cdot s(t)$$

Prin combinația relațiilor de mai sus se obțin legile de comandă de tip sliding-mode:

$$\dot{v}_c = \frac{u_{sm1} - E(1,1) - C_1}{\cos(\gamma_e)}$$

$$\dot{\omega}_c = \frac{u_{sm2} - E(2,1) - C_2}{d \cdot \cos(\gamma_e)}$$

$$C_1 = -\dot{v}_d \cos(\varphi_e + \varphi_d) + v_d \dot{\varphi} \sin(\varphi_e + \varphi_d) + d \dot{\omega}_r \sin(\gamma_e) -$$

$$-(\dot{\theta}_e + \dot{\varphi}) \cdot (v_r \cdot \sin(\gamma_e) - d \cdot \omega_r \cdot \cos(\gamma_e))$$

$$C_2 = \frac{1}{Lh} \cdot [\dot{v}_d \sin(\varphi_e + \varphi_d) + v_d \dot{\varphi} \cos(\varphi_e + \varphi_d) - \dot{v}_r \sin(\gamma_e) -$$

$$-(\dot{\theta}_e + \dot{\varphi}) \cdot (v_r \cdot \cos(\gamma_e) + d \cdot \omega_r \cdot \sin(\gamma_e))] - \dot{\omega}_d$$

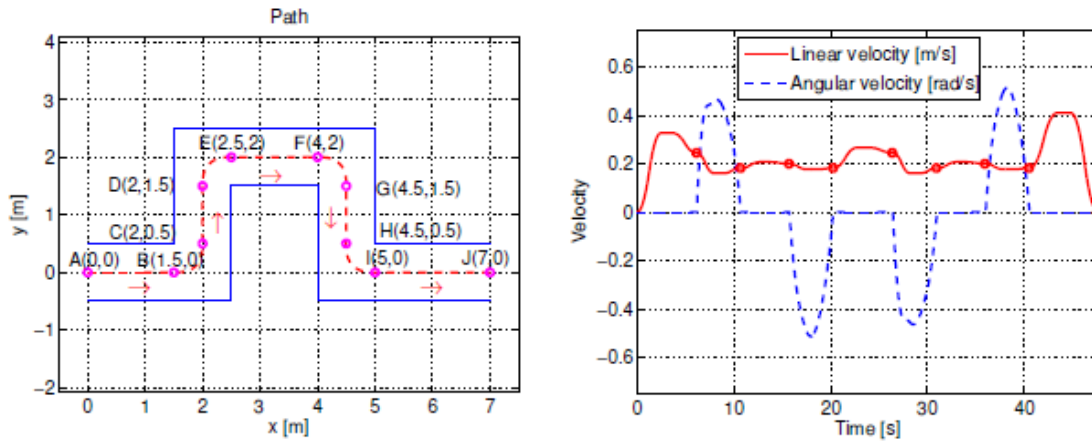


Figura 3. Exemplu de traseu predefinit si vitezele de referinta

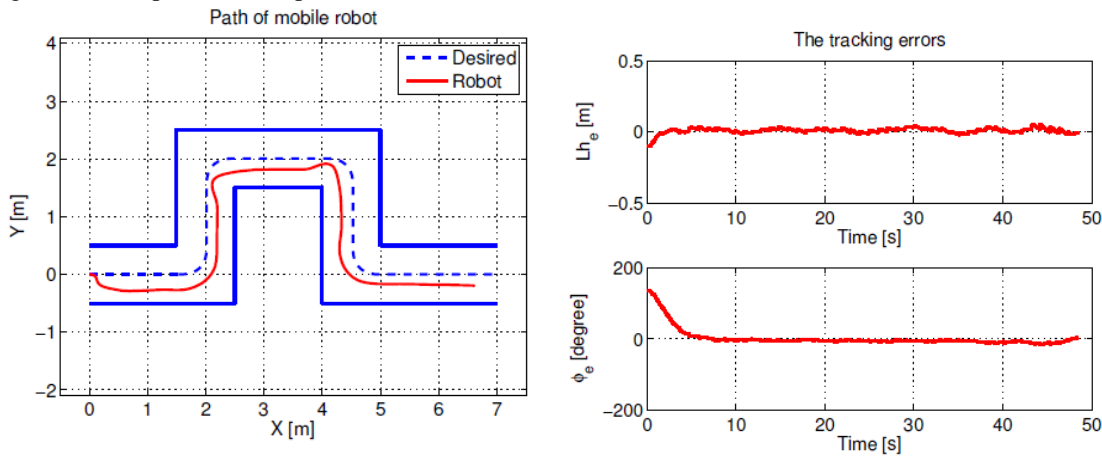


Figura 4. Rezultatele simulării in cazul traseului predefinit si erorile de urmarire

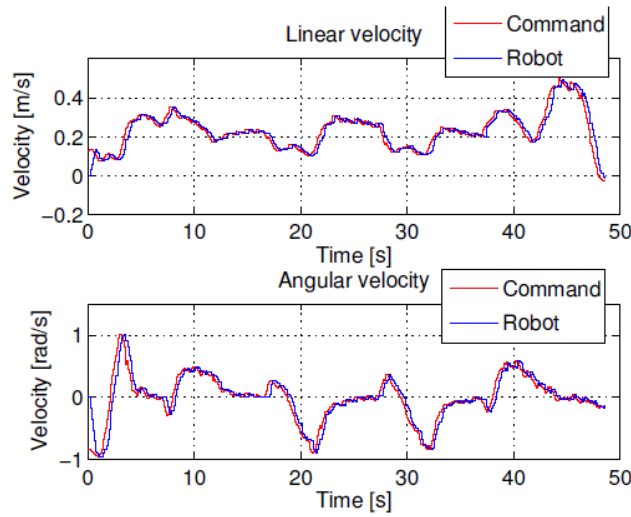


Figura 5. Vitezele liniare si unghiulare (de comanda si simulate)in cazul traseu din Fig. 3.

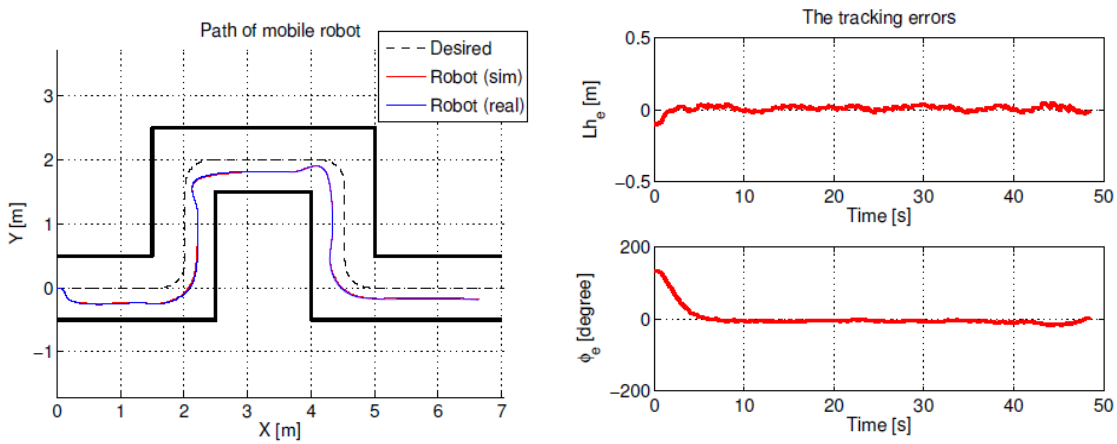


Figura 6. Rezultatele in timp real in cazul traseului predefinit si erorile de urmarire.

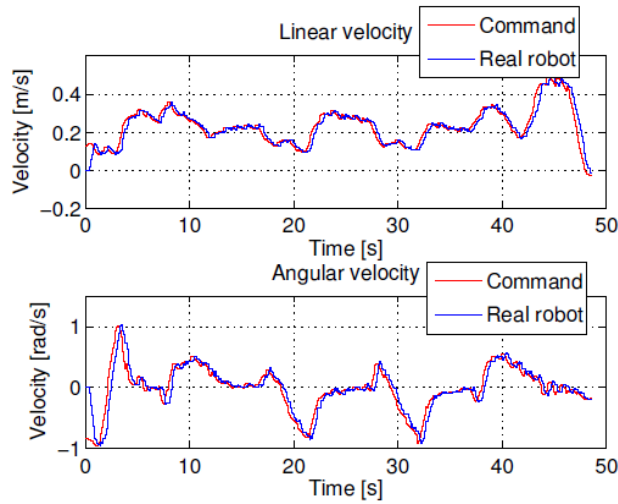


Figura 7. Vitezele liniare si unghiulare (de comanda si reale)in cazul traseu din Fig. 3.

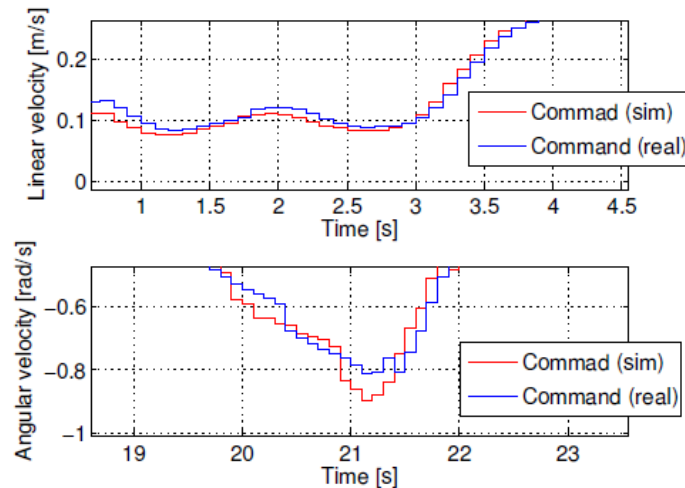


Figura 8. Diferențele dintre vitezele liniare și unghiulare simulate și cele reale.

Activitatea II.1 Proiectarea sistemului de navigație a Prototipului 1 bazată pe combinația dintre sistemul video-biometric al feței și ochiului cu sistemul laser.

1. Resursele hardware: Pentru realizarea sistemului s-au folosit următoarele resurse hardware:

- un scaun electric mobil (Cirrus Power Wheelchair) echipat cu encodere industriale, ce au o rezoluție de 10.000 de impulsuri, pentru cunoașterea poziției și a vitezei în orice moment de timp;

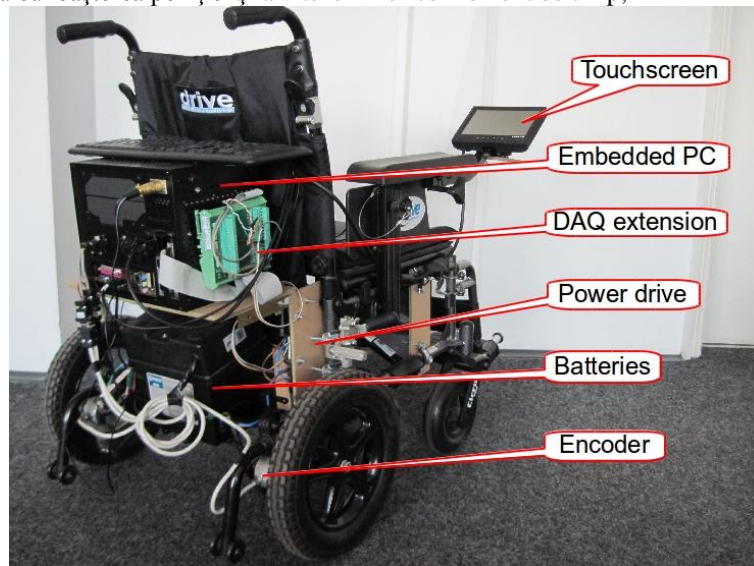


Figura 1. Scaun electric mobil (Cirrus Power Wheelchair) și componentele sale

Sunt utilizate encodere de tip industrial incremental (RI-58) pentru detectarea poziției și a vitezei în orice moment. Caracteristici:

- Rezoluție de 10.000 de impulsuri;
 - Acuratețe mare pentru semnale înalte;
 - Clasa de protecție IP (International Protection Rating) 67;
 - Temperatura de funcționare între -10... +70 grade C și rezistență crescută la șocuri.
- un embedded PC, model Desktop Board ASRock H81M-GL, cu dimensiunile necesare (L: 17.145 x W: 17.145 cm) astfel încât să fie instalat pe scaunul mobil fără a incomoda utilizatorul sau fără a influența comportamentul normal al unui astfel de sistem. De asemenea, computerul este prevăzut cu slot PCI pentru atașarea unei plăci de achiziție model DAQ 6024E.

PC-ul are instalată placă de achiziție și de asemenea programul MATLAB pentru procesarea imaginilor necesare la identificarea direcției. Caracteristicile principale ale PC-ului Desktop Board ASRock H81M-GL sunt următoarele:

- Procesor: CPU Intel 4th generation;

- Memorie: 16 GB RAM;
- Placă grafică dedicată: Sapphire Radeon R5 230 (AMD), 2GB DDR3.

Placa de achiziție DAQ 6024E, este din familia NI 6024E de la National Instruments, este un dispozitiv multifuncțional AI (Analog Input), AO (Analog Output), DIO (Digital Input/Output), pentru calculatoarele dotate cu PCI, PXI sau PCMCIA. Placa are următoarele caracteristici:

- 16 canale AI – cu o rezoluție de 12 biți ;
 - 2 canale AO – cu o rezoluție de 12 biți ;
 - 8 linii TTL – compatibile DIO ;
 - 2 timere/countere DIO pe 24 biți ;
 - Conector I/O cu 68 de pini.
- o camera video low-cost cu senzori CMOS cu sistem de prindere de tip ochelari
 Specificațiile tehnice ale camerei web sunt: tip senzor: CMOS; rezoluție senzor (pixeli): 1.3 MP; rezoluție video (pixeli): 640x480; interfață: USB.

Pentru a surprinde cât mai bine mișcările ochiului s-a recurs la atașarea unei camere web, de o pereche de ochelari. Distanța camerei față de ochi poate fi ajustată astfel încât ochiul, în mod implicit irisul să fie cât mai bine monitorizat și identificat.



Figura 2. Sistemul de prinderea a camerei web de ochelari

2. Descrierea algoritmului

Pentru schema logică a întregului algoritm se va vizualiza Anexa 1.

Pentru a porni camera se folosește funcția start având următoarea formă:

```
start(vid);
```

Pentru a opri camera de folosește următoarea funcție:

```
stop(vid);
```

Legătura dintre MATLAB și camera web se face prin crearea unui obiect de intrare video cu ajutorul următoarei funcții:

```
vid = videoinput('winvideo', 1, 'RGB24_352x288');
```

Stabilirea valorii unei proprietăți a unui obiect video se face cu funcția *set*:

set(vid, 'FramesPerTrigger', 1) -- setează numărul de cadre achiziționate de către *vid* de fiecare dată când se execută o declanșare folosind sursa video selectată.

set(vid, 'TriggerRepeat', Inf) -- setează valoarea declanșării la infinit.

set(vid, 'ReturnedColorSpace', 'rgb') -- setează ca imaginea să fie RGB.

Funcția *triggerconfig(vid, 'Manual')* -- declanșează dispozitivul imediat ce este apelată.

Pentru inițializarea plăcii de achiziție se vor folosi următoarele sintaxe:

```
ao = analogoutput('nidaq','Dev1');
```

crează o ieșire analogică pentru un dispozitiv de la National Instruments definit ca *Dev1*.

După ce a fost creată ieșirea analogică a dispozitivului va trebui să adăugăm canale hardware.

```
aod = addchannel(ao,0) -- se adaugă canalul 0 pentru roata dreapta
```

```
aos = addchannel(ao,1) -- se adaugă canalul 1 pentru roata stângă
```

Funcția $IM=getdata(vid,1,'uint8')$; returnează cadrele asociate dispozitivului video (*vid*) precum și tipul șirului de caractere. În cazul de față 'uint8' reprezintă un număr întreg fără semn pe 8 biți.

Cu ajutorul sintaxei $IM=imcrop(IM,[0,40,320,240])$; se realizează tăierea părților din imagine care ar fi incomodat realizarea unei capturi cât mai bune a ochiului.

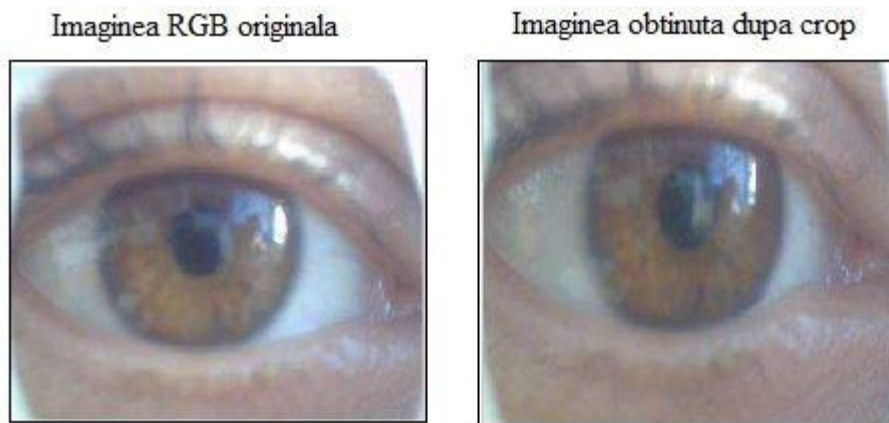


Figura 3. Imaginea obținută după aplicarea funcției crop

$I = rgb2gray(I)$ -- convertește imaginile RGB în imagini de intensitate grayscale prin eliminarea nuanței și a saturației, în timp ce se păstrează luminanța.

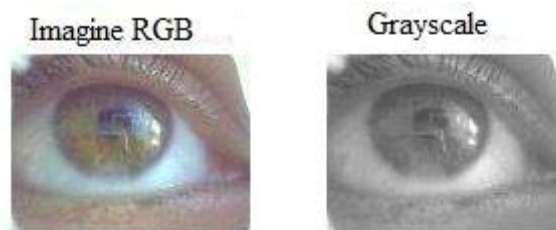


Figura 4. Conversia imaginii RGB în Grayscale

Funcția $imfindcircles(A, radiusRange)$ - găsește cercuri care au razele în intervalul specificat de *radiusRange*. Argumentul radii conține razele aproximative ale cercurilor, iar centers conține coordonatele fiecărui centru al cercului găsit.

```
[center, radii] = imfindcircles(I, [Rmin Rmax], 'ObjectPolarity', 'dark', 'Sensitivity', 0.95)
```

radiusRange -- conține gama de raze pentru obiectele circulare pe care dorim să fie detectate (găsite), specificate ca un vector cu două elemente, [Rmin, Rmax].

ObjectPolarity -- indică dacă obiectele circulare sunt mai luminoase sau mai întunecate decât fundalul imaginii.

'bright' -- indică obiectele mai luminoase;

'dark' -- indică obiectele mai întunecate.

Sensitivity -- reprezintă factorul de sensibilitate. Pe măsura ce crește factorul de sensibilitate, *imfindcircles* detectează obiectele circulare, inclusiv cercurile slabe și parțial acoperite. Dacă valoarea factorului de sensibilitate crește, de asemenea crește și riscul detectării greșite. Factorul de sensibilitate ia valori în intervalul [0, 1].

Funcția *viscircles(centers, radii)* -- desenează cercurile cu centre și raze specificate.

3. Prelucrarea imaginilor și identificarea direcției

Pentru identificarea poziției irisului utilizând cercul înscris în jurul irisului s-a utilizat un algoritm de măsurare a distanței centrului cercului față de axa OX ($0 \div 320$ pixeli) și față de axa OY ($0 \div 240$ pixeli) a imaginii.

S-au definit două benzi de siguranță, una pe axa OY ($60 \div 140$ pixeli) și una pe axa OX cuprinsă în intervalul ($100 \div 170$), cu rolul de a filtra cercurile false care pot apărea în urma mișcărilor involuntare ale ochiului.

Atunci când centrul cercului se află între limitele benzii de siguranță a axei OX mai precis $100 > X_c < 170$, și între limitele benzii de siguranță a axei OY adică $60 > Y_c < 140$ direcția va fi ÎNAINTE.

Dacă centrul cercului are coordonatele cuprinse între $0 > X_c < 100$ și $60 > Y_c < 140$ atunci direcția va fi STÂNGA.

În caz contrar dacă coordonatele centrului cercului sunt cuprinse între $170 > X_c < 320$ și $60 > Y_c < 140$ atunci direcția va fi DREAPTA.

Pentru STOP se va recurge la închiderea ochiului astfel încât să nu fie detectat irisul, ceea ce va duce la oprirea scaunului mobil.

Etapa 1 -- în cadrul acestei etape are loc achiziția imaginilor RGB după ce obiectul video a fost pornit:



Figura 5. Achiziția imaginilor RGB

Etapa 2 -- în această etapă are loc conversia imaginii din RGB în grayscale pentru pregătirea acesteia pentru următoarea etapă.



Figura 6. Imaginea de intensitate grayscale

Etapa 3 -- în cadrul acestei etape se impun benzile de siguranță pentru a filtra cercurile false care pot apărea în urma mișcărilor involuntare ale ochiului.

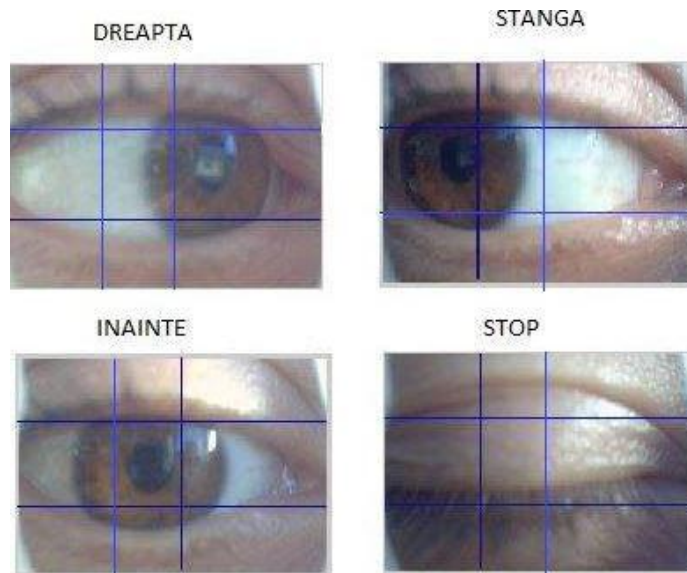


Figura 7. Adăugarea benzilor de siguranță

Etapa 4 -- în această etapă are loc găsirea cercurilor din imagini și evidențierea lor

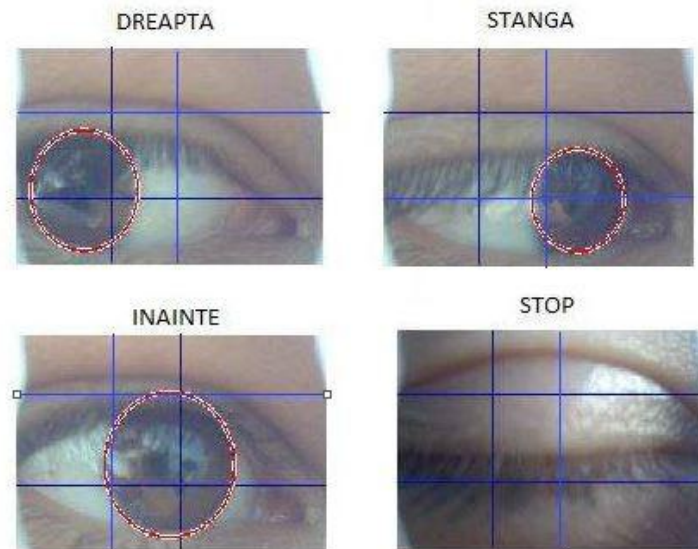


Figura 8. Găsirea cercurilor în imagine și evidențierea lor

Algoritmul propus în lucrarea de față vine în ajutorul persoanelor cu dizabilități locomotorii severe.

Soluția propusă a fost realizată și implementată în timp real în mediul MATLAB și testată pe scaunul electri mobil de tipul Cirrus Power Wheelchair aflat în dotarea Facultății de Automatică, Calculatoare, Inginerie electrică și Electronică.

Pentru implementarea algoritmului s-au utilizat următoarele componente hardware: camera web (Delux DLV-B32) atașată pe o pereche de ochelari, PC integrat pe scaunul electri mobil (Cirrus Power Wheelchair) și o placă de achiziție de tipul DAQ 6024E.

Datorită numărului redus de etape parcurse pentru extragerea irisului și a pupilei din imaginile video achiziționate în timp real, soluția propusă este mult mai rapidă decât alte soluții propuse în literatură.

Sistemul propus permite portabilitatea acestuia pe orice platformă robotică mobilă.

Problemele întâmpinate la implementarea și testarea în timp real au fost următoarele:

- variația intensității luminii;
- performanțele dispozitivului hardware de achiziționare a imaginii (camera CMOS sau CCD);
- reflexia corneei, în special pentru direcția stânga sau dreapta;
- mișcări involuntare ale ochiului. Aceste mișcări au fost rejectate prin benzile de siguranță create, atât pe axa OX cât și pe axa OY, sau prin negăsirea unui număr corespunzător de cercuri pentru a putea fi luat în calcul.

Pentru îmbunătățirea performanțelor algoritmului propus pe viitor: se pot folosi o cameră web mai performantă sau chiar două pentru monitorizarea ambilor ochi (pentru obținerea de informații suplimentare), se poate implementa un sistem de evitare de obstacole de către scaunul cu roțile.

S-a proiectat un algoritm ce permite navigarea scaunului cu roțile într-un mediu dinamic (existând atât obstacole fixe cât și obstacole mobile). Algoritmul preia informația de la senzorul de tip laser (ce poate detecta obstacolele fixe/mobile) și transmite mai departe către scaunul cu roțile (mai exact, către cele două roți motoare) comenzile necesare evitării obstacolelor întâlnite în calea sa.

În această etapă s-a folosit un senzor laser de tip Hokuyo URG-04LX-UG01 (a se vedea figura 9) pentru detecția și evitarea obstacolelor de către un scaun cu roțile până la destinația impusă. Senzorul laser Hokuyo URG-04LX-UG01 este un dispozitiv de scanare și detectare a obstacolelor. Conceput cu un design compact și o greutate foarte mică, având o acuratețe mare este foarte util în aplicațiile cu roboți mobili.

Caracteristicile importante ale senzorului laser sunt: conectivitatea se realizează prin interfața USB 2.0/1.1; aria de detectare este cuprinsă între 20 mm și 5600 mm; acuratețea este de ± 30 mm; unghiul de scanarea se realizează în intervalul 0° - 240° ; cu o rezoluție de $0,35^\circ$ pe fiecare pas, 10 scanări pe secundă; sursa de alimentare: 5v DC $\pm 5\%$, 2,5W.



Figura 9. Senzor laser de tip Hokuyo URG-04LX-UG01

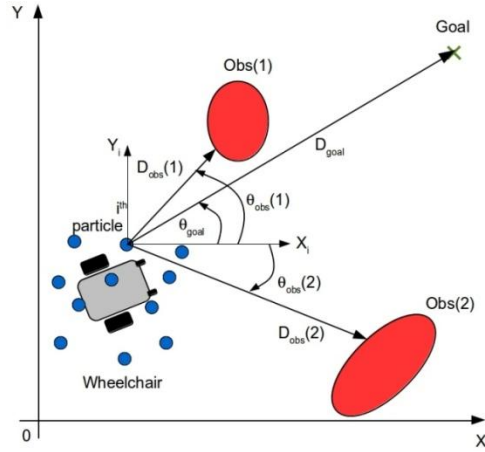


Figura 10. Ilustrarea principiului algoritmului de evitare a obstacolelor fixe si/sau mobile

Se foloseste un ansamblu de particule a căror pozitie sunt din domeniul functiei obiectiv si care sunt modificate printr-un proces iterativ (vezi figura 10). La fiecare iteratie se stabileste noua pozitie a fiecărei particule în functie de:

- Pozitia curentă a particulei;
- Cea mai bună pozitie întâlnită de către particulă (local best);
- Cea mai buna pozitie întâlnită de către ansamblu (global best).

Vectorul pozitiilor particulelor din roi este definit astfel:

$$S_i = \begin{bmatrix} X_i \\ Y_i \end{bmatrix}^T = \begin{bmatrix} x_{i1}, x_{i2}, \dots, x_{in} \\ y_{i1}, y_{i2}, \dots, y_{in} \end{bmatrix}^T$$

Vitezele pe axele X si Y ale particulelor din roi sunt date sub forma vectoriala:

$$V_i = \begin{bmatrix} V_{x_i} \\ V_{y_i} \end{bmatrix}^T = \begin{bmatrix} vx_{i1}, vx_{i2}, \dots, vx_{in} \\ vy_{i1}, vy_{i2}, \dots, vy_{in} \end{bmatrix}^T$$

Cea mai bună pozitie a particulei i si cea mai bună pozitie a ansamblului de particule:

$$P_i = \begin{bmatrix} P_{x_i} \\ P_{y_i} \end{bmatrix}^T = \begin{bmatrix} px_{i1}, px_{i2}, \dots, px_{in} \\ py_{i1}, py_{i2}, \dots, py_{in} \end{bmatrix}^T; P_g = \begin{bmatrix} P_{gx_i} \\ P_{gy_i} \end{bmatrix}^T = \begin{bmatrix} px_{g1}, px_{g2}, \dots, px_{gn} \\ py_{g1}, py_{g2}, \dots, py_{gn} \end{bmatrix}^T$$

Regula de ajustare a pozitie particulelor:

$$\begin{cases} V_i^{k+1} = WV_i^k + c_1r_1(P_i - S_i^k) + c_2r_2(P_g - S_i^k) \\ S_i^{k+1} = S_i^k + V_i^k \end{cases} \quad (1)$$

Funcția fitness folosită pentru evitarea obstacolelor fixe și mobile este:

$$Fitness = K_{goal} \cdot D_{goal} + K_{\theta_{goal}} \cdot |\theta_{goal}| + K_{obs} \cdot e^{-K_{D_{obs}} \cdot D_{obs}(m)} \cdot |\theta_{obs}(m)|^{-K_{\theta_{obs}}} \quad (2)$$

TABLE I
PATH PLANNER ALGORITHM.

```

1: Initialization Parameters:
   iterations, swarm size, initial swarm position, initial velocity.
2: for iter = 1 to iterations do
3:   for i = 1 to swarm size do
4:     update x position (see eq. 1)
     update y position (see eq. 1)
5:     calculate the fitness (see eq. 2)
6:     if new position is better then
       update best x
       update best y
       update the best value
7:     end if
8:   end for
9: update global best positioning
10:  for i = 1 to swarm size do
11:   updating velocity vectors (see eq. 1)
     x velocity components
     y velocity component
12:  end for
13: end for

```

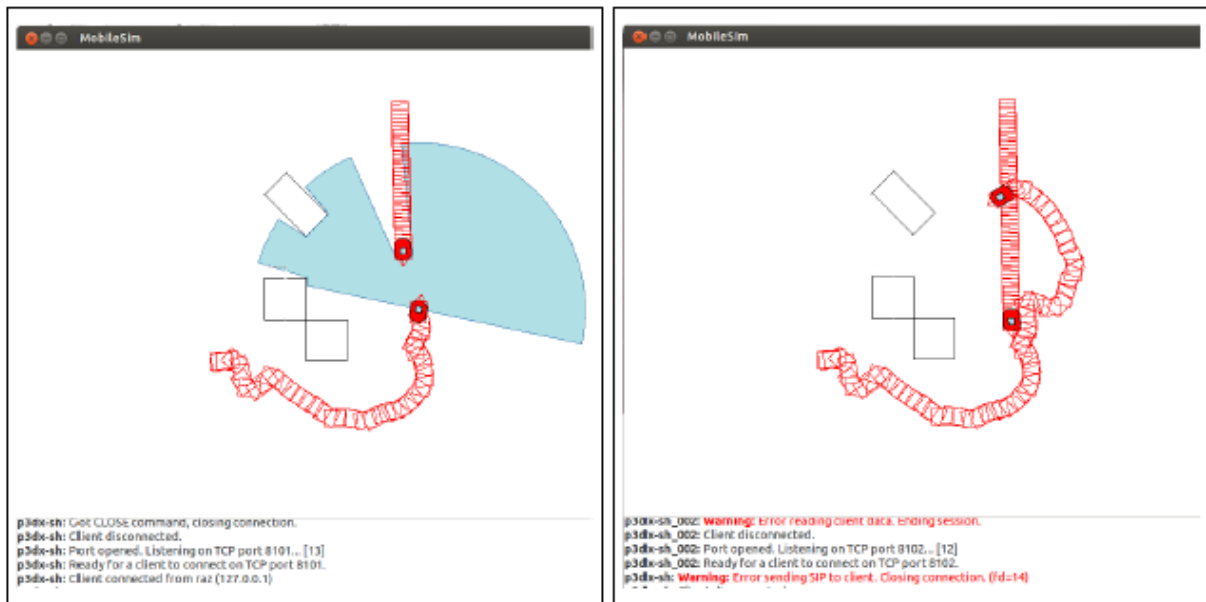


Fig. 11. Rezultate obținute după rularea algoritmului (local) propus utilizând mediile de programare ARIA și MobilSime.

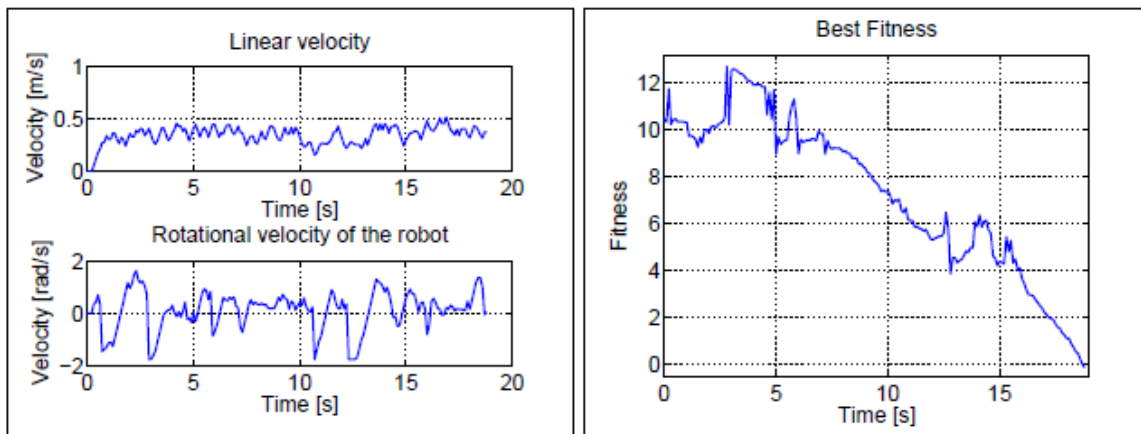
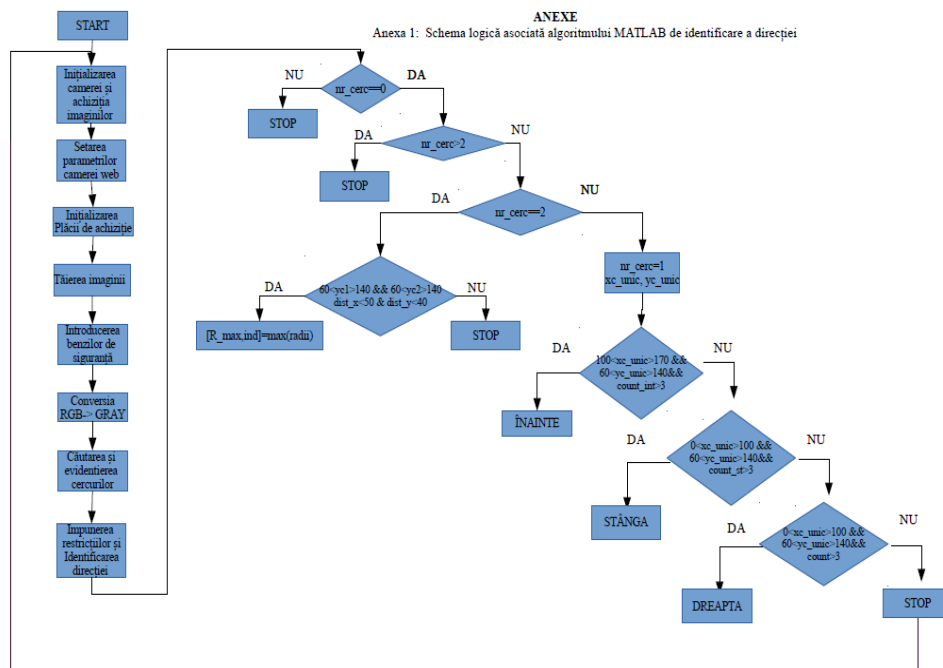


Fig. 12. Vitezele liniare și unghiulare ale robotului mobil precum și cele mai bune valori ale funcției fitness obținute în urma simulărilor.



Activitatea II.2:Proiectarea structurii de conducere sliding-mode și backstepping, pentru urmărirea unei traiectorii impuse, bazată pe modelul cinematic al Prototipului 2. Proiectarea sistemului de navigație a Prototipului 2 bazată pe ultrasunete și laser.

Prototipul 2 (Fig.1) include o platformă mobilă Pioneer P3-DX, un modul de comunicație cu conversie RS232 la WiFi, un manipulator robotic AL5D pentru PLTW Lynxmotion cu modul bluetooth și un controler NI-CVS-1454.

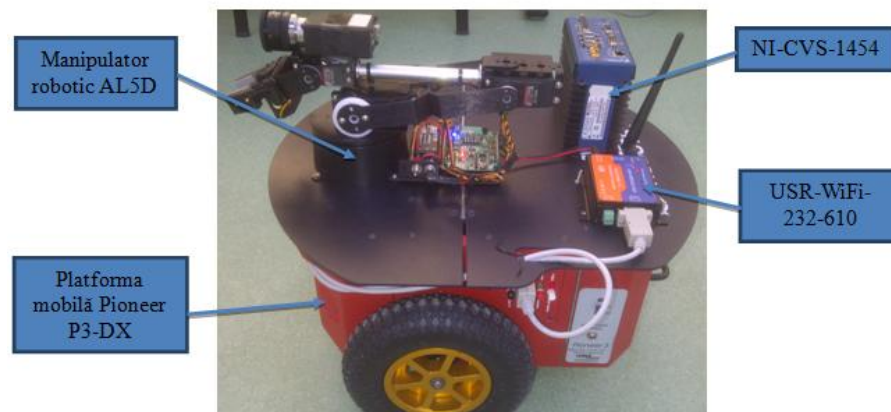


Figura 1. Prototip 2

Platforma mobilă Pioneer P3-DX (Fig. 2) face parte dintr-o familie de roboți mobili produși de compania MOBILE ROBOTS. În aceasta familie intrând roboți Pioneer 1, Pioneer AT, Pioneer 2-DX, și multi alți. Aceste platforme mobile de dezvoltare și cercetare au în comun arhitectura și software-ul, fiind echipate cu sistem de conducere pe 2 sau 4 roți.

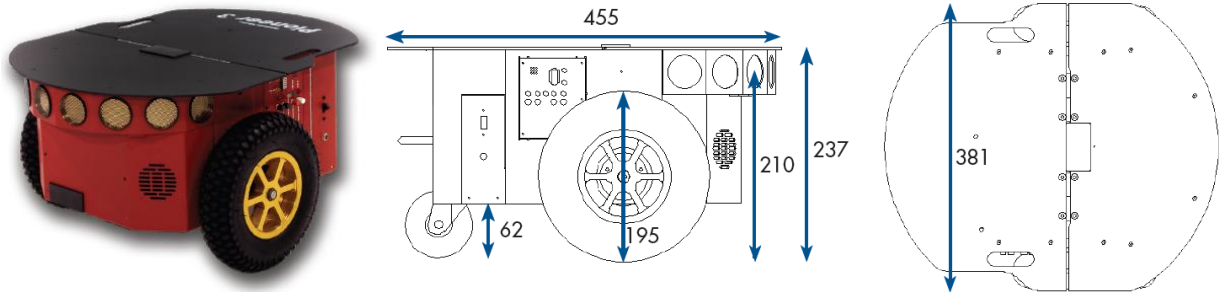


Figura 2. Robot mobil Pioneer P3-DX

Pioneer P3-DX este dotat cu sistem de conducere onboard, devenind astfel un robot mobil autonom. Spre deosebire de alți roboți, dimensiunea mică a platformei mobile Pioneer P3-DX permite navigarea pe culoare înguste și spații aglomerate. Sistemul de conducere al robotului Pioneer 3-DX folosește două motoare de curent continuu, fiecare fiind echipat cu un encoder optic de înaltă rezoluție pentru o poziționare precisă și o sensibilitate ridicată la determinarea vitezei.

Pioneer P3-DX poate urca în rampă cu o înclinare de maximum 25%, pe teren plan viteza robotului mobil poate ajunge până la 1,6 m/s (5,76 km/h). Cântărește 9 kg cu un număr minim de baterii. Aceste caracteristici îi permit să transporte o sarcină de pînă la 23kg. Robotul mobil este echipat în parte frontală cu un sistem de senzori de tip de sonare. Cele 8 sonare au rază de acțiune cuprinsă între 15cm și 5m.

Poziționarea sonarelor este realizată prin dispunerea a câte un sonar pe fiecare parte a robotului din cele 8 iar celelalte 6 în fața, situate la un interval de 15° unul de celălalt. Opțional poate dispune și de un inel de sonare în partea din spate, cu aceeași configurație. Pe partea de control dispune de un calculator embedded PC/104 și module de I/O prin care se pot conecta diverse dispozitive externe.

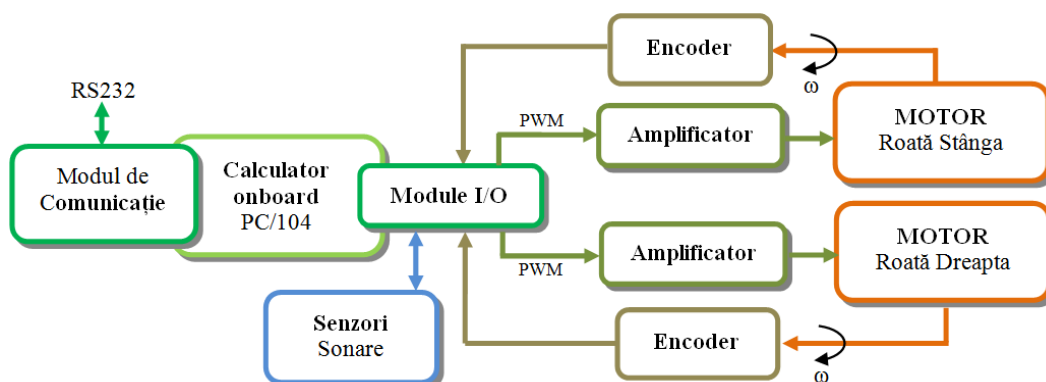


Figura 3. Schema bloc a sistemului de conducere încorporat în Pioneer P3-DX

Manipulatorul robotic AL5D pentru PLTW Lynxmotion (Fig. 4) oferă mișcări rapide, precise și repetabile.

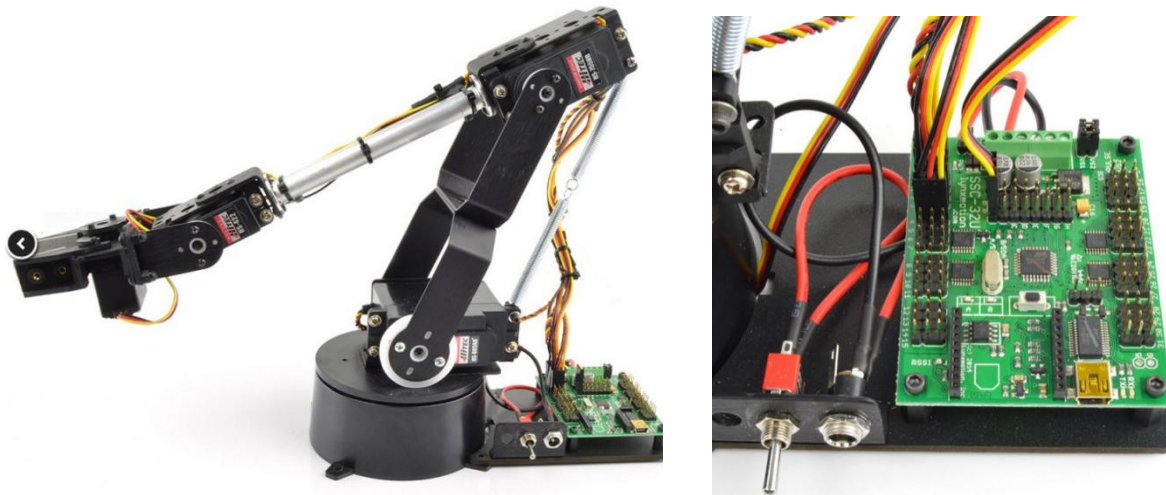


Figura 4. Manipulatorul robotic AL5D

Caracteristicile sale includ - umăr într-un singur plan, cot, încheietura mâinii, un gripper funcțional, și opțional rotirea încheieturii mâinii. Brațul folosește mai multe servomotoare, 1 x HS-485HB în baza, 1 x HS-805BB în umăr, 1 x HS-755HB în cot, 1 x HS-645 mg în încheietura mâinii, și 1 x HS-422 în elementul de prindere. Rotirea se face cu un unghi de 180 grade. Placa de comandă este SSC-32U aceasta putând comanda până la 32 servomotoare cu rezoluție de 1us și accesa 8 canale I/O analogice. Folosește un microcontroler Atmel ATMEGA328P cu memorie EEPROM de 512 biți ce poate comunica la distanță fie prin USB, Xbee, UART sau N81. Rata de bit poate să fie de 9600, 38.4kbps și 115.2 kbps selectabilă prin butoane sau programarea registrilor. Se ofera 32 canale PWM, interfața spre PC fiind de tip mini USB. Pentru programare se recomandă FlowBotics Studio.

Convertorul RS232/RS485 la WiFi / Ethernet este utilizat pentru a transmite comenzile dinspre PC / Laptop oferite spre un COM, prin TCP/IP, spre portul serial COM al robotului, precum și pentru a popula baza de date de pe PC/Laptop cu date obținute în activitatea robotului și manipulatorului.

Am ales un produs USR IOT Technology (Fig. 5) care este programat cu funcție de server în sistema, astfel încât poate fi conectat direct la modulul WiFi al PC-ului sau la un AP din rețeaua instituției / întreprinderii. Standardul corespunde lui 802.11 b/g/n, interfața UART operând la debite cuprinse între 1200 și 230400 bps iar modulul Ethernet la 100Mbps. Suportă comenzi AT și securitate tip WEP și WPA. Modulul 485 respectiv portul de Ethernet un sunt folosite în aplicația noastră.



Figura 5. Convertorul RS232/RS485 spre WiFi/Ethernet

Sistemul de viziune compactă NI-CVS-1454 (Fig. 6) produs de National Instruments vine să substituie calculatorul PC104 al robotului, care din motive financiare nu a putut fi achiziționat. Sistemul a fost achiziționat într-un proiect CEEX și este funcțional. În această etapă a proiectului nu este pus în valoare deoarece conducerea robotului se face prin comenzi expediate prin RS232 - TCP/IP – RS232 de la PC / Laptop iar brațul robotic este acționat de la distanță prin bluetooth din aplicația proprie producătorului. Intenția este ca în etapa următoare să se permită rularea comenzilor de mișcare ale robotului și brațului robotic dintr-o aplicație gestionată de CVS, în plus putând prin prelucrări de imagini, obținute de la o cameră video sau chiar un sistem stereoscopic, să ajute mișcările brațului robotic și chiar mișcarea robotului.



Figura 6. Sistemul de viziune compact NI-CVS-1454

Sistemul beneficiază de o memorie internă este de 128MB DRAM fiind alimentat la 12V. Are 13 canale de intrare izolate, 2 TTL, 4 canale de ieșire izolate, 10 TTL, 1 port serial, 3 porturi Firewire(IEEE 3194) operând la 100fps cu rezoluție 2000x2000, 1 port Ethernet de 100Mbps, 1 port VGA iar programarea se poate face utilizând LabView Real-Time Module.

Proiectarea structurii de conducere sliding-mode

Arhitectura conducerii sliding-mode în timp continuu a WMR cu 2 DW/1FW este prezentată în Fig. 7. Pentru calculul comenzii considerăm un vehicul virtual care se deplasează pe traiectoria dorită. Problema urmăririi traiectoriei presupune proiectarea unui controler care să calculeze viteza liniară și viteza unghiulară care să permită vehiculului real să urmărească traiectoria vehiculului virtual cu erori de poziționare cât mai mici.

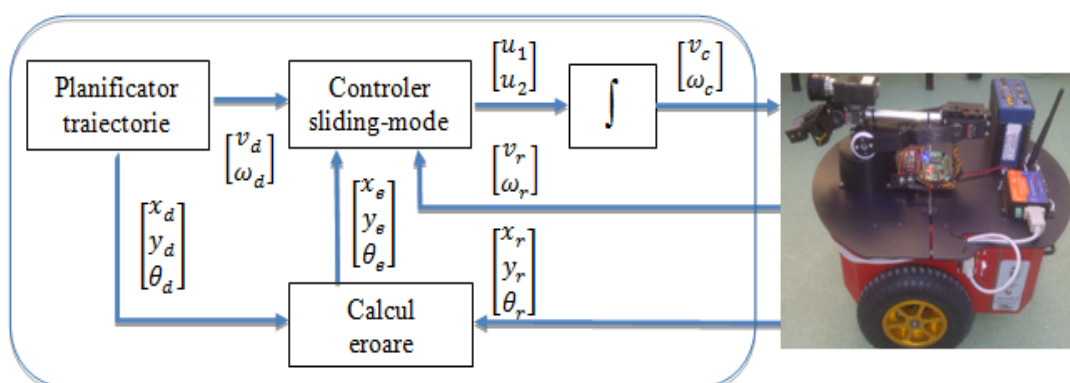


Figura 7. Arhitectura conducerii sliding-mode pentru WMR 2DW/1FW

Modelul cinematic al vehiculului virtual a fost prezentat în etapa 1, acesta fiind:

$$\begin{cases} \dot{x}_d = v_d \cdot \cos \theta_d \\ \dot{y}_d = v_d \cdot \sin \theta_d \\ \dot{\theta}_d = \omega_d \end{cases} \quad (\text{II.2.1})$$

unde (x_d, y_d) reprezintă coordonatele carteziene ale centrului geometric, v_d este viteza liniară, θ_d reprezintă orientarea și ω_d reprezintă viteza unghiulară.

Erorile de urmărire sunt:

$$\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos \theta_d & \sin \theta_d & 0 \\ -\sin \theta_d & \cos \theta_d & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_r - x_d \\ y_r - y_d \\ \theta_r - \theta_d \end{bmatrix} \quad (\text{II.2.2})$$

Dinamica erorilor de urmărire este:

$$\begin{cases} \dot{x}_e = -v_d + v_r \cdot \cos \theta_e + \omega_d \cdot y_e \\ \dot{y}_e = v_r \cdot \sin \theta_e - \omega_d \cdot x_e \\ \dot{\theta}_e = \omega_r - \omega_d \end{cases} \quad (\text{II.2.3})$$

Erorile de urmărire descrise de (II.2.2) sunt prezentate în Fig. 8.

Considerând erorile de urmărire și derivata lor (II.2.3) se definesc suprafețele de comutație:

$$s_1 = \dot{x}_e + k_1 \cdot x_e \quad (\text{II.2.4})$$

$$s_2 = \dot{y}_e + k_2 \cdot y_e + k_0 \cdot \text{sign}(y_e) \cdot \theta_e \quad (\text{II.2.5})$$

Legea de control este folosită în forma:

$$\dot{s} = -Q \cdot \text{sgn}(s) - P \cdot s \quad (\text{II.2.6})$$

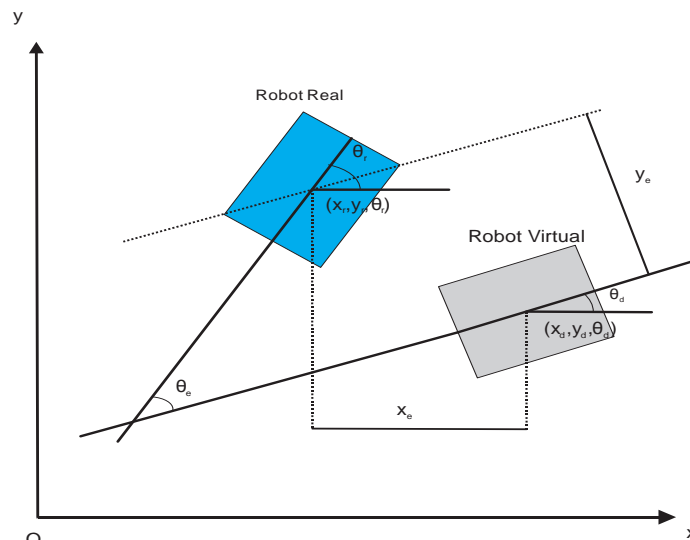


Figura 8. Erorile de urmărire în cazul roboților mobili

Putem obține astfel legea de comandă sliding mode:

$$\dot{v}_c = \frac{-Q_1 \cdot \text{sign}(s_1) - P_1 \cdot s_1 - k_1 \cdot \dot{x}_e - \dot{\omega}_d \cdot y_e - \omega_d \cdot \dot{y}_e + v_r \cdot \dot{\theta}_e \cdot \sin \theta_e + \dot{v}_d}{\cos \theta_e} \quad (\text{II.2.7})$$

$$\omega_c = \frac{-Q_2 \cdot \text{sign}(s_2) - P_2 \cdot s_2 - k_2 \cdot \dot{y}_e - \dot{v}_c \cdot \sin \theta_e + \dot{\omega}_d \cdot x_e + \omega_d \cdot \dot{x}_e}{v_r \cdot \cos \theta_e + k_0 \cdot \text{sgn}(y_e)} + \omega_d \quad (\text{II.2.8})$$

Fie $V = \frac{1}{2} \cdot s^T \cdot s$ o funcție Lyapunov. Derivata acestei funcții este:

$$\dot{V} = s_1 \cdot \dot{s}_1 + s_2 \cdot \dot{s}_2 = s_1 \cdot (-Q_1 \cdot \text{sgn}(s_1) - P_1 \cdot s_1) + s_2 \cdot (-Q_2 \cdot \text{sgn}(s_2) - P_2 \cdot s_2) \quad (\text{II.2.9})$$

Derivata funcției Lyapunov este negativă semi-definită dacă alegem parametrii $Q_i, P_i \geq 0$.

Controlerul propus pentru conducerea WMR 2DW/1FW a fost testat în simulări realizate cu ajutorul softului MobileSim de la Mobile Robots respectiv în mediu real. Programul a fost scris în C++ și rulat pe un laptop pentru o frecvență de eșantionare de 100 ms. În continuare sunt prezentate simulări și teste în timp real pentru a valida metodele propuse. Pentru a trimite comenzile calculate de algoritmi de conducere s-au folosit funcțiile din ARIA: setVel(Velocity), setVel2(leftVelocity, RightVelocity), setRotVel(Angular Velocity). S-au citit datele de la robot cu ajutorul funcțiilor ARIA: getX(), getY(), getTH(), getRotVel(), getVel().

În primul test s-a dorit urmărirea unei traiectorii liniare cu viteza de 0.5m/s de către vehiculul 2DW/1FW utilizând conducerea sliding-mode în timp continuu (10 m, 20 s).

În Fig. 9 este prezentată simularea în MobileSim a traiectoriei vehiculului autonom Pioneer 2DW/1FW utilizând conducerea sliding-mode în timp continuu. Se observă în figurile care urmează, rezultatele din *simulator* (cu linie roșie continuă) iar și cele corespunzătoare conducerii în *timp-real* (cu linie albastră întreruptă) după cum urmează:

- Fig. 10 prezintă traiectoria rezultată în urma conducerii sliding-mode în timp continuu;
- Fig.11 prezintă eroarea pe axa Ox obținută prin simularea conducerii sliding-mode în timp continuu;
- Fig.12 prezintă eroarea pe axa Oy generată de simularea conducerii sliding-mode în timp continuu; Fig. 13 prezintă eroarea de orientare a robotului în cazul conducerii sliding-mode în timp continuu;
- Fig.14.a prezintă suprafața de alunecare s_1
- Fig. 14.b prezintă suprafața de alunecare s_2 calculate în timpul conducerii sliding-mode în timp continuu a vehiculului.

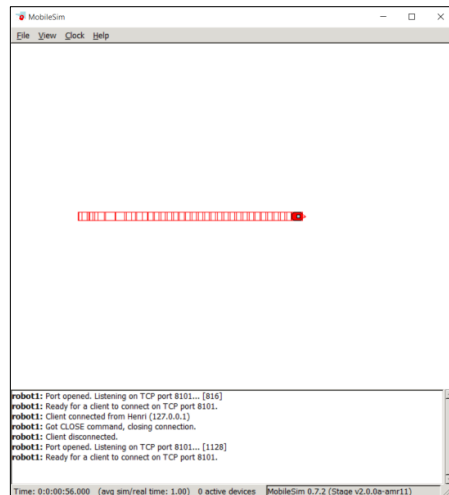


Figura 9. Conducerea sliding-mode în timp continuu (test 1). Traectoria simulată în MobileSim

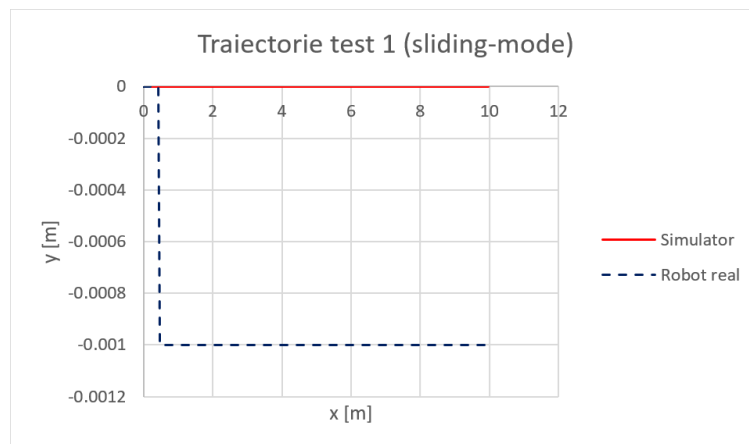


Figura 10. Conducerea sliding-mode în timp continuu (test 1). Traiectorie în simulator și reală

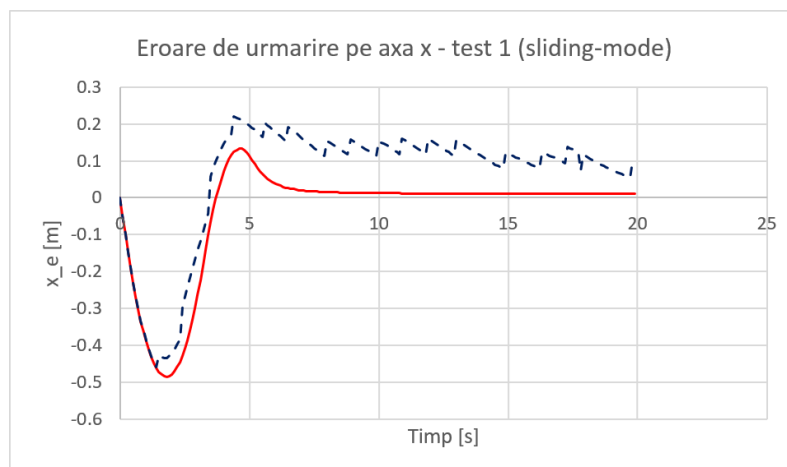


Figura 11. Conducerea sliding-mode în timp continuu (test 1). Eroarea de urmărire pe axa X.

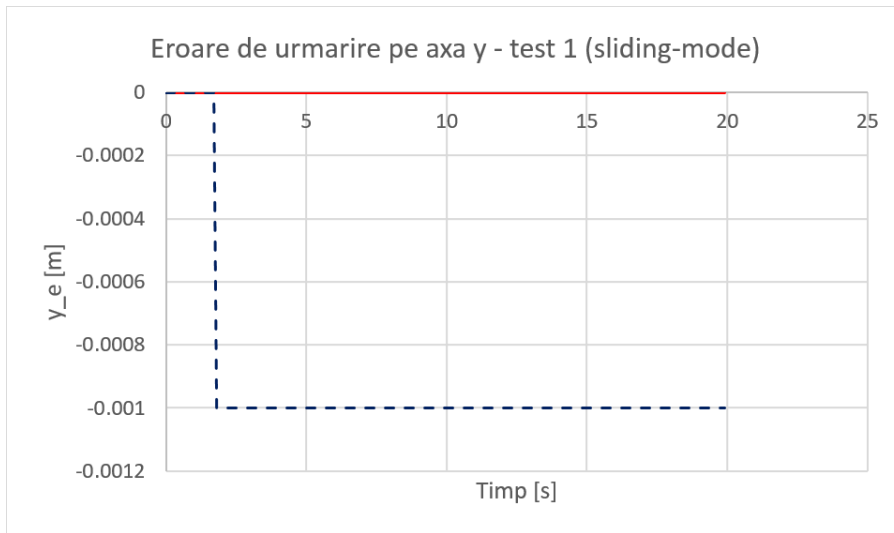


Figura 12. Conducerea sliding-mode în timp continuu (test 1). Eroarea de urmărire pe axa Y

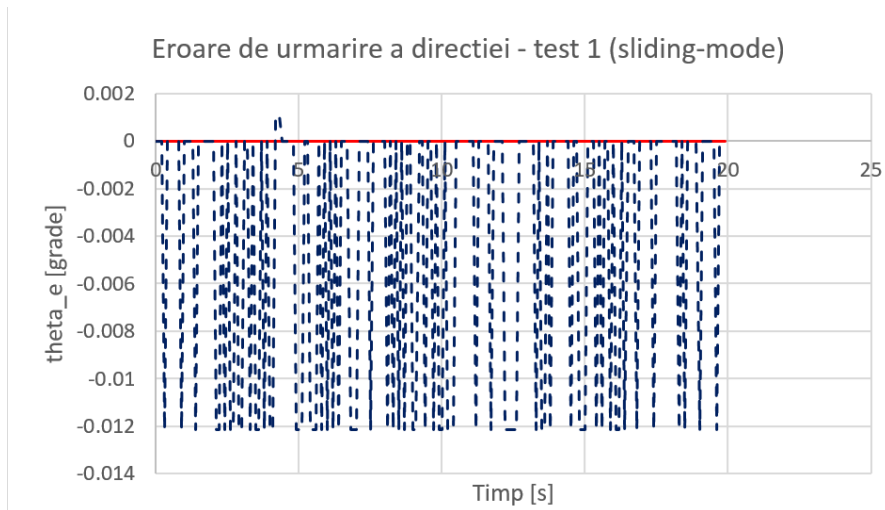
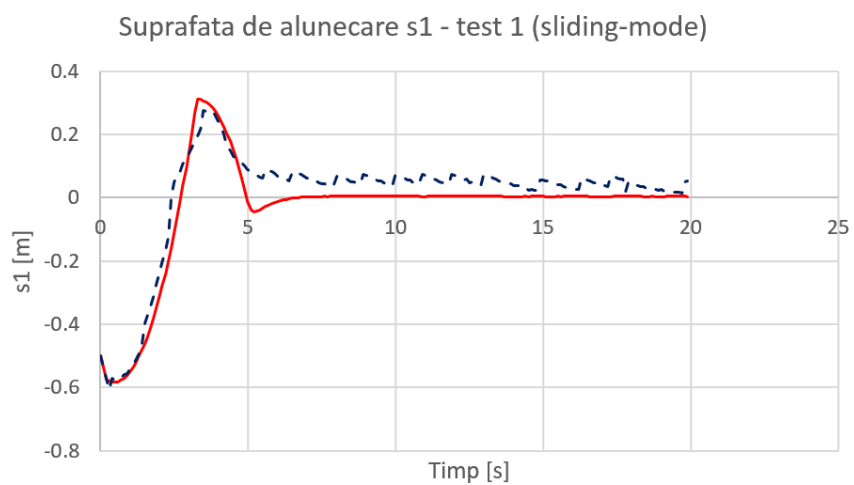


Figura 13. Conducerea sliding-mode în timp continuu (test 1). Eroarea de urmărire a direcției



a)

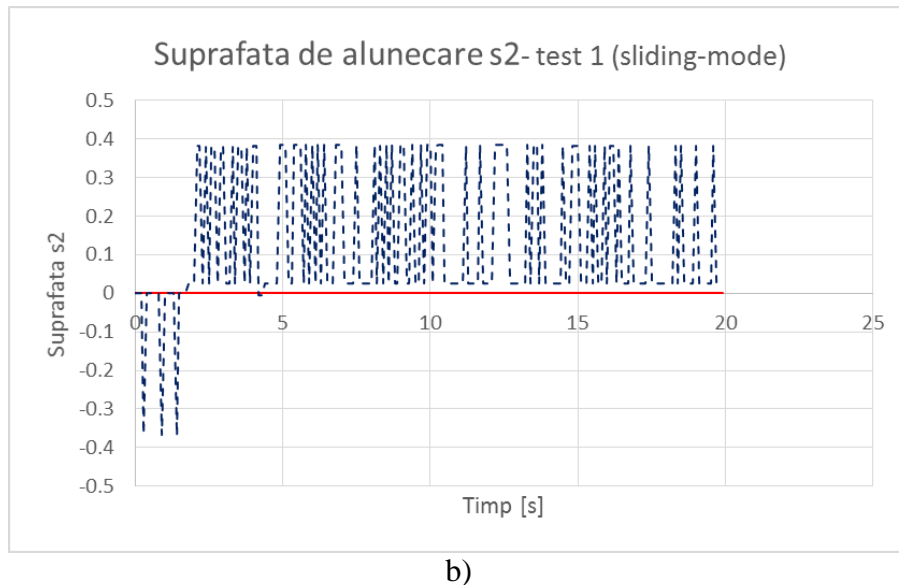


Figura 14. Test 1. a) Suprafața de alunecare s_1 ; b) Suprafața de alunecare s_2

Se observă că erorile de poziționare sunt mici în zona liniară pentru primii 10m respectiv primele 20s, performanțele de urmărire a traiectoriei fiind destul de bune în cazul folosirii conducerii sliding-mode în timp continuu pentru WME 2DW/1FW.

Pentru control s-au folosit $Q_1=0.05$, $Q_2=0.5$, $P_1=0.5$, $P_2=0.75$ (în legea de control), $k_0=30$, $k_1=0.75$, $k_2=25$ (pentru suprafețele de comutație), $v_d = 0.5\text{m/s}$, 200 de iterații în linie dreaptă ($w=\omega_d=\phi_d'=0$).

În al doilea test s-a dorit urmărirea unui traseu impus, corespunzător unuia dintr-o hală de producție (definit în etapa I a proiectului), de către vehiculul 2DW/1FW utilizând conducerea sliding-mode în timp continuu (parcurgele pentru 13 linii drepte). Este reluat controlul sliding-mode pentru fiecare segment al traseului, imaginile evidențiind aceiași parametri ca în primul test, aceștia fiind cumulați pentru întreg traseul.

Astfel în Fig. 15 este prezentată simularea în MobileSim a traiectoriei vehiculului autonom WMR Pioneer 2DW/1FW utilizând conducerea sliding-mode în timp continuu. Se observă în figurile care urmează rezultatele din simulator (linie roșie continuă) după cum urmează:

- Fig. 16 prezintă traiectoria rezultată în urma conducerii sliding-mode în timp continuu;
- Fig. 17 prezintă eroarea de orientare a robotului în cazul conducerii sliding-mode în timp continuu;
- Fig. 18 prezintă eroarea pe axa O_x obținută prin simularea conducerii sliding-mode în timp continuu;
- Fig. 19 prezintă eroarea pe axa O_y generată de simularea conducerii sliding-mode în timp continuu;
- Fig. 20 prezintă suprafața de alunecare s_1 calculată în timpul conducerii sliding-mode în timp continuu a vehiculului.

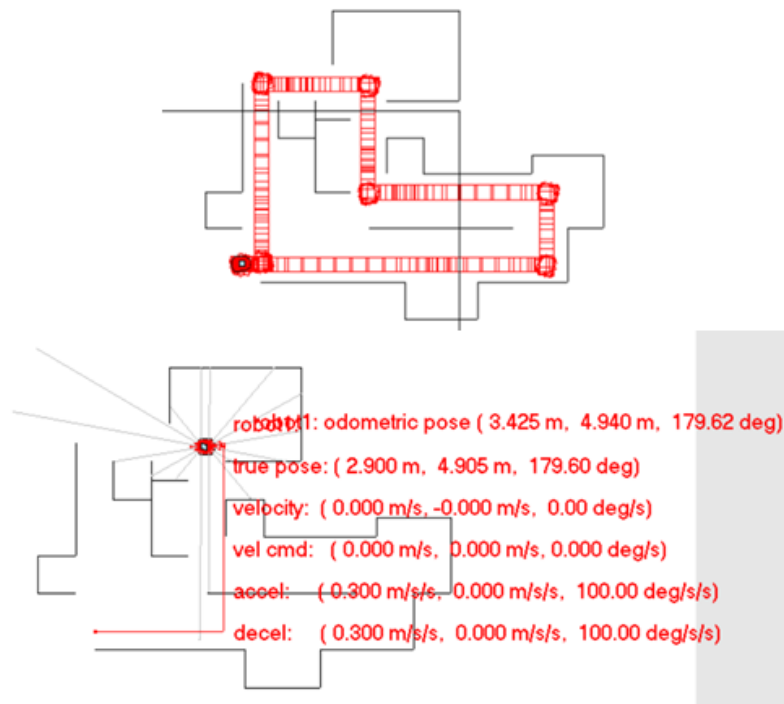


Figura 15. Conducerea sliding-mode în timp continuu (test 2). Traectoria simulată în MobileSim

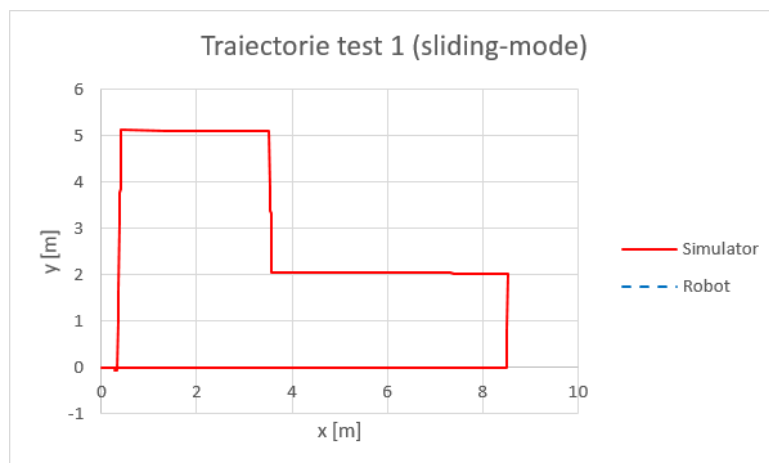


Figura 16. Conducerea sliding-mode în timp continuu (Test 2). Traiectorie în simulator

Pentru control s-au folosit $Q_1=0.05$, $Q_2=0.5$, $P_1=0.5$, $P_2=0.75$ (în legea de control), $k_0=30$, $k_1=0.75$, $k_2=25$ (pentru suprafețele de comutație), $v_d = 0.5\text{m/s}$, 200 de iterații în linie dreaptă ($w=\omega_d=\phi_d'=0$).

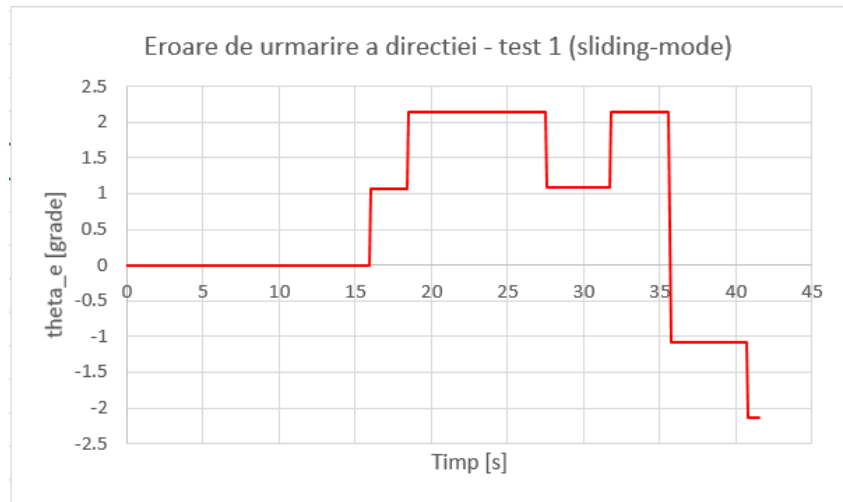


Figura 17. Conducerea sliding-mode în timp continuu (Test 2). Eroarea de urmărire a direcției

În figura 17 se observă schimbările de direcție cu 90° , 180° , 90° , 180° , 270° și 180° (tranzițiile) respectiv erorile foarte mici de urmărire a direcției prin utilizarea sliding-mode, în liniile drepte (palier). În figurile 18 și 19, cât timp sliding-mode a fost aplicată după fiecare punct de întoarcere și după fiecare operație de stop în dreptul unui punct de lucru, putem remarca efectul blocului de integrare din Fig.7 de fiecare dată când robotul începe deplasarea (Fig. 18) respectiv eroarea minimă (Fig. 19) pe axa y pentru fiecare linie dreaptă, orizontală în Fig.15.

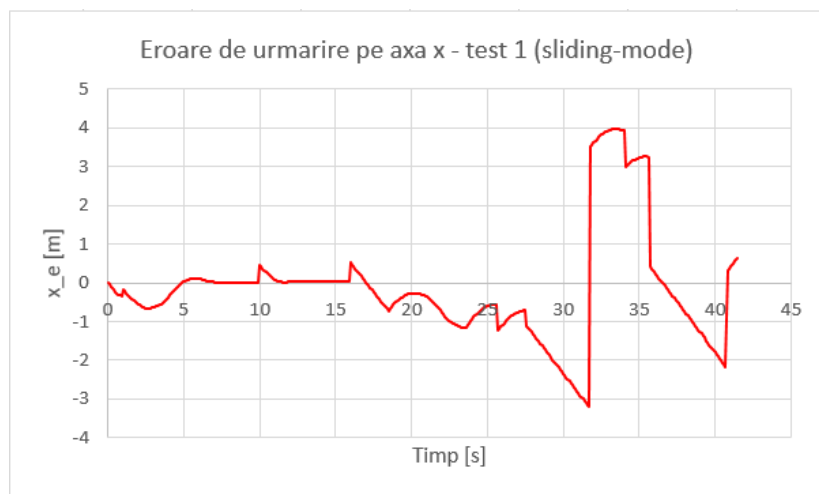


Figura 18. Conducerea sliding-mode în timp continuu (test 2). Eroare de urmărire pe axa X.

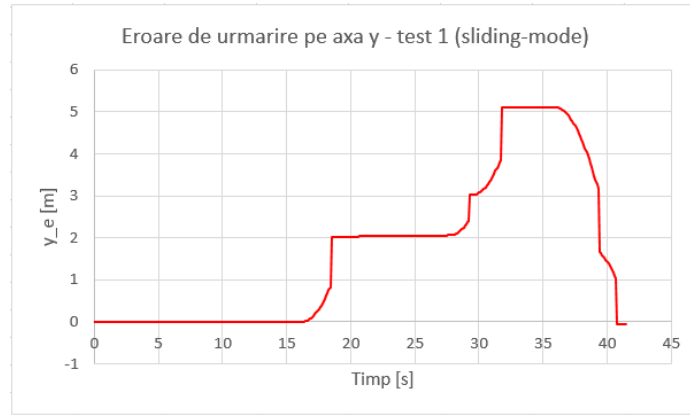


Figura 19. Conducerea sliding-mode în timp continuu (test 2). Eroare de urmărire pe axa Y

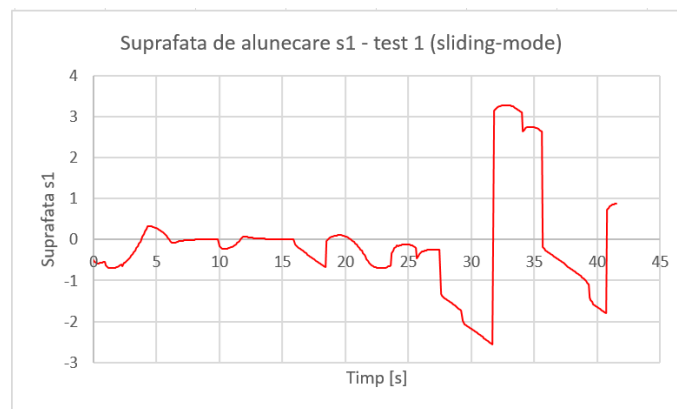


Figura 20. Test 2 - Suprafata de alunecare s1

Proiectarea structurii de conducere backstepping

Pentru a calcula comanda în cazul roboților mobili pentru problema urmăririi globale, considerăm modelul cinematic al robotului 2DW/1FW (Fig. 7), modelul cinematic al robotului virtual (Fig. 8) erorile de urmărire (II.2.2) și dinamica erorilor de urmărire (II.2.3).

Conducerea backstepping presupune calculul legilor de control ale vitezelor liniare și unghiulare având forma:

$$v = v(x_e, y_e, \theta_e, v_d, \omega_d, \dot{v}_d, \dot{\omega}_d) \quad (\text{II.2.10})$$

$$\omega = \omega(x_e, y_e, \theta_e, v_d, \omega_d, \dot{v}_d, \dot{\omega}_d) \quad (\text{II.2.11})$$

astfel încât traiectoria în buclă închisă definită de de (II.2.3), (II.2.10), (II.2.11) să fie mărginită și să convergă spre zero.

Integratorul backstepping se folosește pentru că y_e nu este controlat direct în (II.2.3). Funcțiile $x_e = k_1 \cdot \omega_d \cdot y_e$ și $\theta_e = 0$ sunt funcții stabilizatoare pentru y_e în (II.2.3).

Folosim schimbarea de variabilă:

$$\bar{x}_e = x_e - k_1 \cdot \omega_d \cdot y_e \quad (\text{II.2.12})$$

unde $k_1 > 0$ este o constantă.

Utilizând (II.2.13) în (II.2.3) se obține ecuația:

$$\dot{\bar{x}}_e = -v_d + v_r \cdot \cos \theta_e + \omega_d \cdot y_e - k_1 \cdot \dot{\omega}_d \cdot y_e - k_1 \cdot \omega_d \cdot (-\omega_d \cdot x_e + v_r \cdot \sin \theta_e) \quad (\text{II.2.13})$$

Folosind schimbarea de variabilă:

$$u = v_r \cdot \cos \theta_e + \omega_d \cdot y_e - k_1 \cdot \omega_d \cdot (-\omega_d \cdot x_e + v_r \cdot \sin \theta_e) \quad (\text{II.2.14})$$

Rezultă comanda backstepping a roboților mobili:

$$v = u - y_e \cdot \omega_d + k_2 \cdot \bar{x}_e \quad (\text{II.2.15})$$

$$\omega = \omega_r + \lambda \cdot y_e \cdot v_r \int_0^1 \cos(s \cdot \theta_e) ds + k_3 \cdot \lambda \cdot \theta_e \quad (\text{II.2.16})$$

Arhitectura de conducere backstepping este prezentată în Fig. 21.

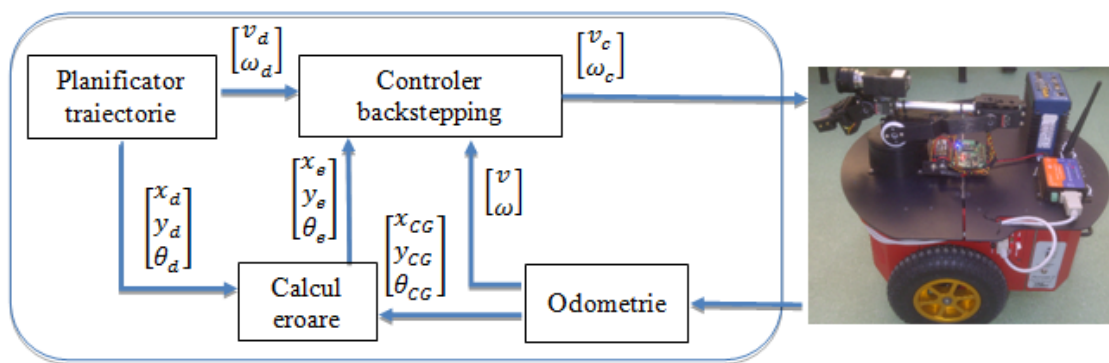


Figura 21. Arhitectura de conducere backstepping

Activitatea II.3: Modelarea, utilizand Retele Petri Hibrade(RPH), a liniilor mecatronice de asamblare/dezasamblare (Hara&Horstman)si deprelucrare/reprelucrare (Festo) deservite de roboti autonomiechipati cu manipuloare (sisteme robotice autonome din gama Prototipului 2).

II.3.1. Linia mecatronică de asamblare/dezasamblare HERA & Horstmann



Figura 1. Sistemul flexibil de asamblare/dezasamblare HERA & Horstmann

Linia mecatronica HERA & Horstmann (Fig.1) este compusă din 5 stații (celule) de asamblare, depozitul de produse finite, un sistem de transport de tip benzi transportoare precum și sisteme de manipulare și depozitare de tip lift (Fig. 6.a). Stațiile de prelucrare S1, S2, S3 și S4 (Fig. 2.a, Fig. 5) au asociată câte o magazie de piese, în fiecare magazie găsiindu-se una din componentele P1, P2, P3, P4 care se asamblează în produsului final (Fig.2.b).

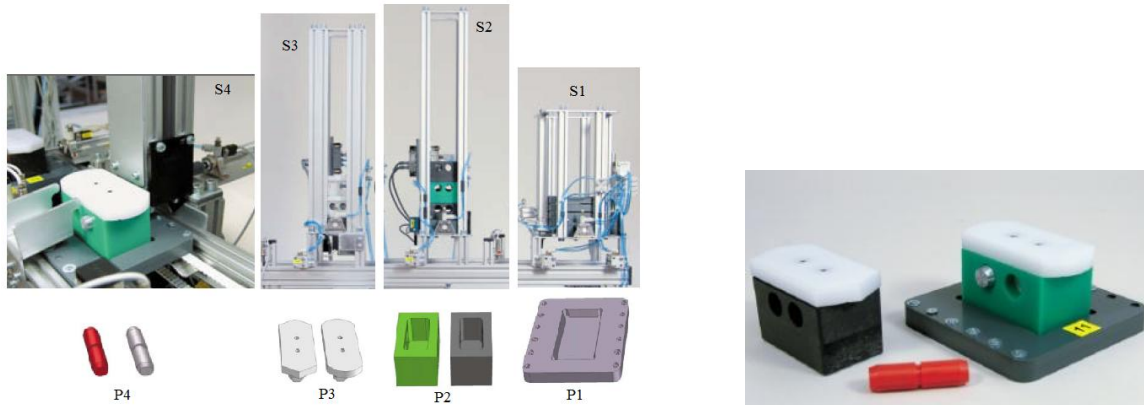


Figura 2. a) Stațiile de asamblare/dezasamblare și componentele aferente acestora; b) Produs final

Depunerea unei piese pe banda transportoare se realizează prin intermediul unor elemente de execuție de tip piston pneumatic, acționat de un sistem pneumatic. Fiecare stație este echipată cu traductori de poziție, pentru o localizare precisă în dreptul magaziei corespunzătoare.

Structura sistemului de automatizare (Fig.3) este de tip distribuită și este formată dintr-un AP SIEMENS Simatic S7-300 cu procesor din seria CP 314C-2 DP și modul de comunicație SIEMENS CP 343-2. Acesta se conectează pe magistrala PROFIBUS DP care conectează modulele auxiliare MA de interfațare I/O de tip SIEMENS ET200S-IM 151-1 distribuite pe fiecare dintre stațiile sistemului flexibil de ansamblare/dezasamblare. Fiecare din cele 6 module SIEMENS ET200S-IM 151-1 prezintă module de I/O digitale și analogice, acestea preluând semnale provenite de la traductoare și transmițând comenzi elementelor de execuție

Pe magistrala PROFIBUS DP este conectat un terminal de tip panou operator SIEMENS Simatic HMI TP 177, prin intermediul căruia se poate vizualiza starea sistemului flexibil și se poate pune în execuție un proces de asamblare sau dezasamblare.

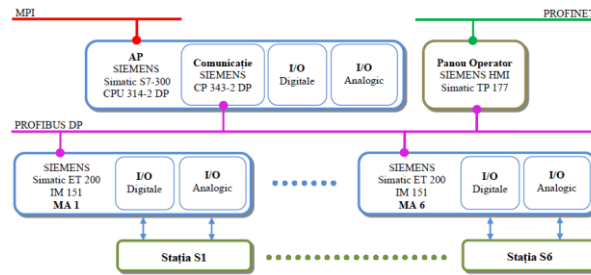


Figura 3. Arhitectura sistemului de automatizare al liniei flexibile de asamblare/dezasamblare HERA & Horstmann

Pentru modelarea sistemului flexibil este necesară identificarea taskurilor elementare ale proceselor de asamblare și dezasamblare. Operația de dezasamblare se declanșează dacă un produs este calificat ca fiind rebut în urma testului de analiză a calității. Prin dezasamblare se vor recupera toate componentele piesei asamblate. Acestea se depun în magaziile aferente de către robotul mobil care le preia din stațiile de dezasamblare (Fig. 5. a, b). Linia mecatronică devine astfel un sistem automatizat total reversibil în care fiecare asamblare este urmată fie de o dezasamblare - atunci când piesa asamblată este rebut, fie de o nouă asamblare.

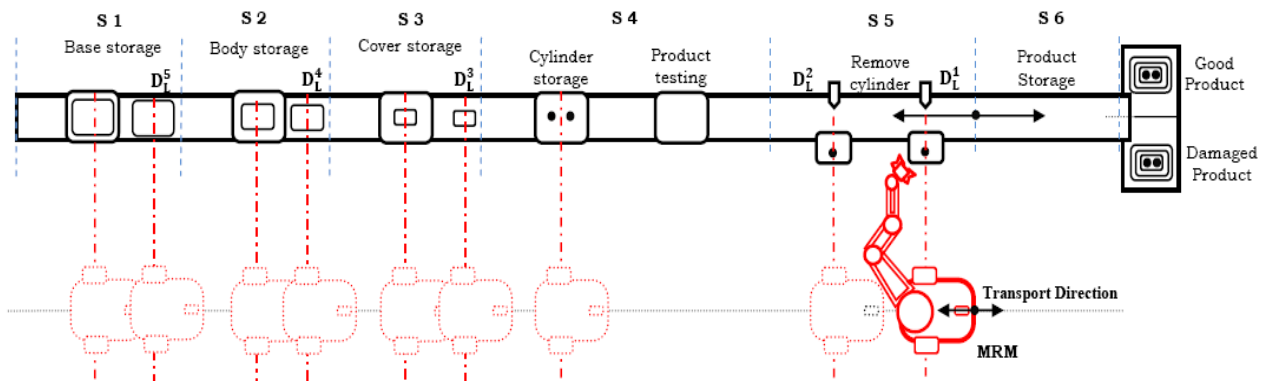


Figura 4. Linia mecatronică de asamblare/dezasamblare a unei piese cu 5 componente, deservită de un robot mobil echipat cu manipulator.

II.3.2. Modelarea cu SHPN a liniei mecatronice HERA & Horstmann

Dinamica liniei A/DML deservită de roboți mobili echipați cu manipuloare este descrisă printr-o tipologie SED dacă se consideră doar operațiile de asamblare și dezasamblare. Sistemul de deservire reprezentat de roboți mobili echipați cu manipuloare aduce în modelul general SHPN variații ale stărilor procesului determinate de evenimente dar și de variația continuă în timp a unor parametri ai acestuia. Modelul global are o structură ierarhică (Selinger, *et al.*, 1999; Radaschin, 2011; Radaschin *et al.*, 2012; Kopacek, and Kopacek, 1999) care utilizează tipologii RP temporale discrete dar și tipologii RP hibride și cu sincronizare. Fiecare operație de asamblare/dezasamblare/deservire este reprezentată printr-un modul RP elementar: e-TPN (corespunde unei operații de asamblare/depozitare în magazie), e-SPN+TPN (corespunde unei operații de dezasamblarea unui produs rebut), e-THPN (corespunde acțiunilor de deservire a unei secvențe de dezasamblare de către RM echipat cu manipulator).

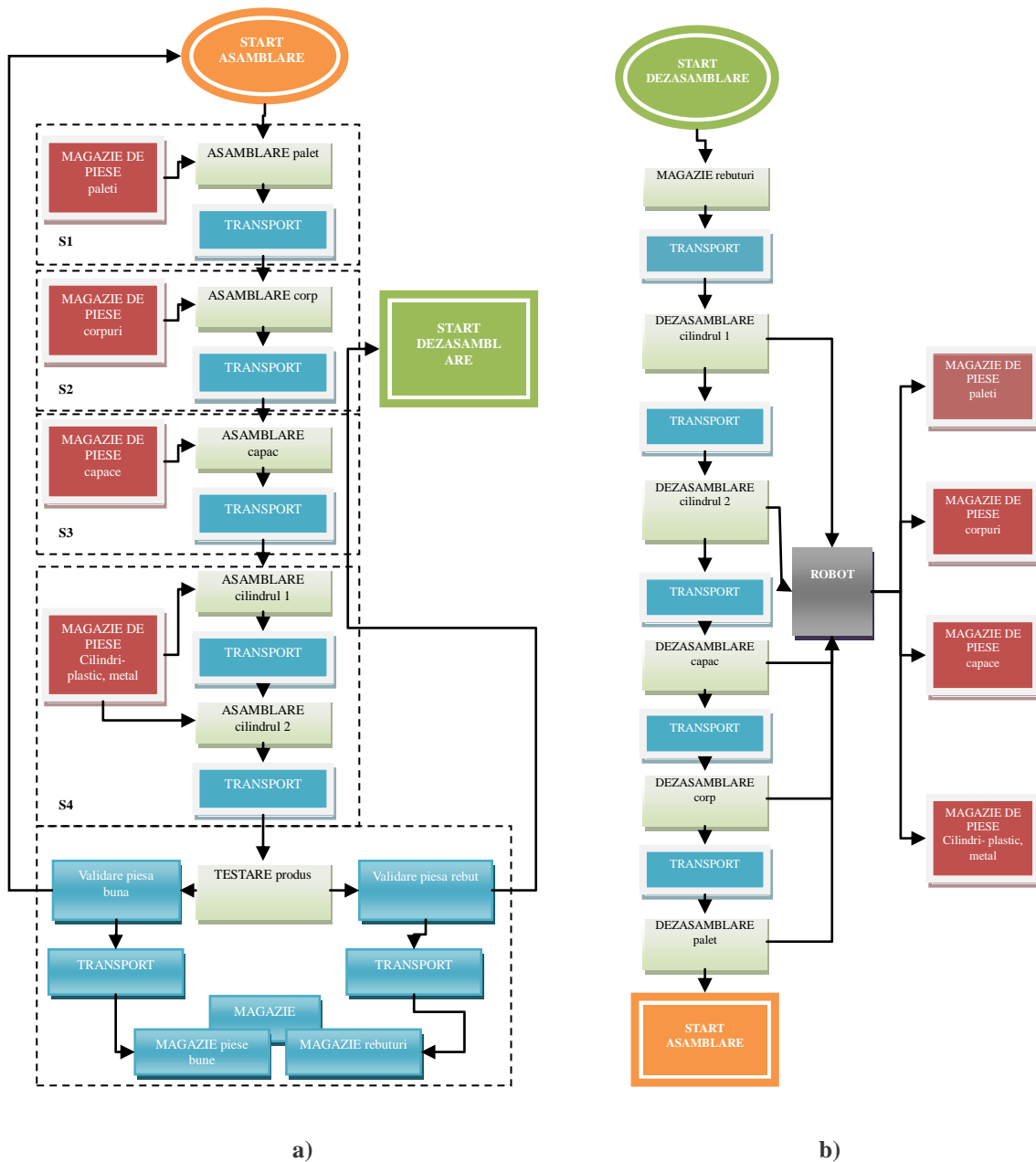


Figura 5 .Schema bloc a operațiilor de asamblare/dezasamblare pe linia mecatronică HERA & Horstman

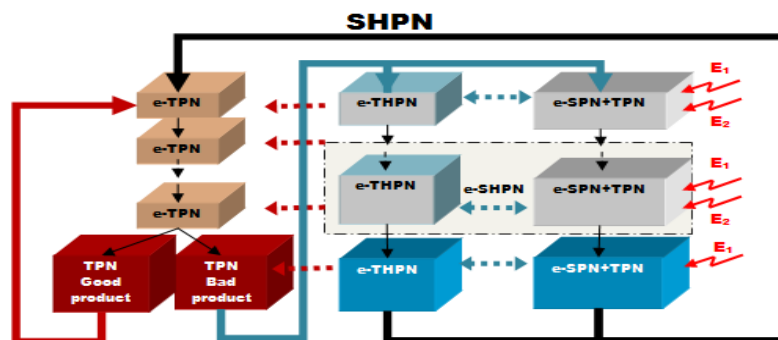


Figura 6. Structura modulară a modelului SHPN corespunzător liniei mecatronice de asamblare/dezasamblare deservită de RM echipat cu manipulator.

Modelul SHPN este interfațat cu semnale externe de sincronizare care validează tranzițiile la care acestea sunt asociate, în timpul simulării on-line și în controlul în timp-real al procesului.

$E_{dd(j)}^1$ și $E_{dd(j+2)}^2$ sunt semnale achiziționate de la senzori, utilizate în sincronizarea liniei cu RM echipat cu manipulator. $E_{dd(j)}^1$ este semnal extern de sincronizare corespunzător acțiunilor: STOP linie și START dezasamblare. $E_{dd(j+2)}^2$ este semnal extern de sincronizare corespunzător acțiunilor: PICKING UP componentă dezasamblată și START linie.

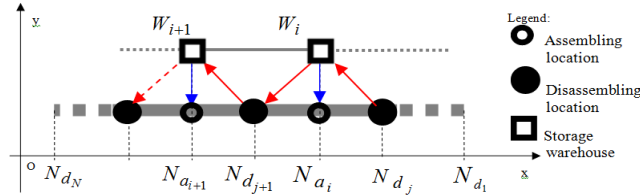


Figura 7. Reprezentarea generală a locațiilor unui flux de N activități de asamblare/dezasamblare

În conformitate cu Fig.7 se consideră următoarele notații:

- $N_{a_i}, i = \overline{1, N}$ - locațiile operațiilor de asamblare în sensul pozitiv al axei Ox ;
- $N_{d_j}, j = \overline{1, N}$ - locațiile operațiilor de dezasamblare în sensul negativ al axei Ox ;
- $W_i; i = \overline{1, N}$ - locațiile magaziiilor de componente;
- $D(N_{d_j}, W_{N+1-j})$ - distanța dintre locația dezasamblării N_{d_j} și magazia aferentă W_{N+1-j} ;
- $D(W_{N+1-j}, N_{d_{j+1}})$ - distanța dintre magazia W_{N+1-j} și locația de dezasamblare următoare $N_{d_{j+1}}$;
- $D_{r_j} = D(N_{d_j}, W_{N+1-j}) + D(W_{N+1-j}, N_{d_{j+1}})$ - distanța parcursă de RM în cadrul unei secvențe de dezasamblare;
- $r = 1 + (j-1) \cdot 3$ - indexează stările continue ale RM - Pcr ; a continuous transition of the robot, Tcr sunt tranzițiile cu viteză continuă asociate deplasării RM, Tdd sunt tranzițiile asociate operațiilor de dezasamblare;
- $k = 1 + (j-1) \cdot 5$ - indexează stările discrete Pdd asociate stării produsului dezasamblat.
- $l = 1 + (j-1) \cdot 4$ - indexează tranzițiile asociate acțiunilor discrete ale RM, Tdr .

II.3.3. Modelul SHPN corespunzător liniei mecatronice HERA & Horstmann

$$\underline{SHPN} = \langle THPN, E, Sync \rangle \quad (1)$$

unde:

$$THPN = \langle P, T, Pre, Post, m_0, h, tempo \rangle \quad (2)$$

$$E = \left\{ Edd_i^1, Edd_j^2 \right\}_{\substack{i=1+3(k-1) \\ j=3(k-1), k=\overline{1, N}}} \cup \{e\} \quad (3)$$

$$Sync : T \rightarrow \{E^1, E^2\} \cup \{e\} \quad (4)$$

$$\text{Sync} : \{Tdd_r\}_{r=1+3(k-1), k=1, \overline{N}} \rightarrow \{E^1, E^2\} \quad (5)$$

$$\text{Sync} : \{Tdd_i\}_{i=3(k-1), k=2, \overline{N}} \rightarrow \{Edd_i^2\}_{k=3(k-1), k=2, \overline{N}}$$

$$\text{Sync} : T \setminus \{Tdd_r\}_{r=1, 3+3(N-1)} \cup \{Tdr_l\}_{l=1, 4+5(N-1)} \cup \{Tcr_r\}_{r=1, 3+3(N-1)} \rightarrow e$$

$$\underline{\text{THPN} = \langle P, T, Pre, Post, m_0, h, tempo \rangle}$$

$$P = \{P_1, P_2, \dots, P_n\} = P^D \cup P^C \quad (6)$$

$$P^D = \{Pda_i\}_{i=1, 13+4(N-1)} \cup \{Pdd_r\}_{r=1, 5+5(N-1)} \cup \{Pdr_s\}_{s=1, 4+8(N-1)}$$

$$P^C = \{Pcr_k\}_{k=0, 3+3(N-1)}, \quad (7)$$

unde (Fig.9.):

$\{Pda_i\}$ mulțimea stărilor discrete (Fig.9.a);

$\{Pdd_j\}$ mulțimea stărilor discrete asociate piesei în curs de dezasamblare (Fig.9.b; Fig.9.c)

$\{Pdr_k\}$ mulțimea stărilor discrete asociate robotului mobil (RM) în timpul acțiunilor de deservire;

$\{Pcr_k\}$ mulțimea stărilor continue asociate distanțelor parcurse de RM în timpul depășirii între postul de dezasamblare și magazinele de depozitare;

$$T = \{T_1, T_2, \dots, T_m\} = T^D \cup T^C \quad (8)$$

$$T^D = \{Tda_i\}_{i=1, 7+2N} \cup \{Tdd_r\}_{r=1, 3+3(N-1)} \cup \{Tdr_l\}_{l=1, 4+5(N-1)} \quad (9)$$

$$T^C = \{Tcr_r\}_{r=1, 3+3(N-1)}$$

$\{Tda_i\}$ mulțimea tranzițiilor discrete asociate operațiilor de asamblare;

$\{Tdd_j\}$ mulțimea tranzițiilor discrete asociate operațiilor de dezasamblare;

$\{Tdr_k\}$ mulțimea tranzițiilor discrete asociate acțiunilor de manipulare ale RM, în cursul operațiilor de dezasamblare;

$\{Tcr_k\}$ mulțimea tranzițiilor continue asociate acțiunilor de deplasare ale RM în cursul operațiilor de dezasamblare. Aceste tranziții au asociate în simularea modelului SHPN viteze lineare maxime;

$Pre : P \times T \rightarrow Q_+$ or N mulțimea arcelor de intrare;

$Post : P \times T \rightarrow Q_+$ or N mulțimea arcelor de ieșire;

$m_0 : P \rightarrow R_+$ or N marcaj inițial;

$$h : P \cup T \rightarrow \{D, C\} \quad (10)$$

$$h : P^D \cup T^D \rightarrow \{D\}$$

$$h : P^C \cup T^C \rightarrow \{C\} \quad (11)$$

tempo este funcția de temporizare a tranzițiilor :

d_{da_i} reprezintă durata operației “i” de asamblare;

d_{dd_r} reprezintă durata operației “r” de dezasamblare;

d_{dr_i} reprezintă durata acțiunii “l” a RM pentru manipularea piesei dezasamblate;

$$tempo : T \rightarrow Q_+ \cup \{0\} \quad (12)$$

$$\text{If } T_j \in T^D, T_a^D = \{Tda_i\}_{i=2, \overline{k=1, N}} \cup \{Tda_{2(N+1)}\}$$

$$tempo(Tda_i) = d_{da_i} \quad (15)$$

$$T_d^D = \{Tdd_r\}_{r=1+3(k-1), \overline{k=1, N}}, \quad (16)$$

$$tempo(Tdd_r) = d_{dd_r}$$

$$T_r^D = \{Tdr_i\}_{i=4+5(k-2), \overline{k=2, N}}, \quad (17)$$

$$tempo(Tdr_i) = d_{dr_i}$$

$$\text{If } T_{cr} \in T_C \text{ atunci } U_r = \frac{1}{tempo(T_{cr})} \quad (18)$$

Pentru $N = 5$ (A/DML Hera&Horstmann) (Fig.8) și considerând toate convențiile de notare, modelul SHPN devine :

$$\underline{SHPN} = \langle THPN, E, Sync \rangle$$

$$Sync : \{Tdd_j\}_{j=\{1,3,4,6,7,9,12,13\}} \rightarrow \{Edd^1, Edd^2\}$$

unde:

$$Sync : \{Tdd_i\}_{i=\{1,4,7,13\}} \rightarrow \{Edd_i^1\}_{i=\{1,4,7,13\}}$$

$$Sync : \{Tdd_i\}_{i=\{3,6,9,12\}} \rightarrow \{Edd_i^2\}_{i=\{3,6,9,12\}}$$

$$Sync : \{Tdd_j\}_{j=\overline{1,16}} \cup \{Tdr_k\}_{k=\overline{1,21}} \cup \{Tcr_k\}_{k=\overline{1,10}} \rightarrow e$$

$$\underline{THPN} = \langle P, T, Pre, Post, m_0, h, tempo \rangle$$

$$P^D = \{Pda_i\}_{i=\overline{1,29}} \cup \{Pdd_j\}_{j=\overline{1,25}} \cup \{Pdr_k\}_{k=\overline{1,41}}$$

$$P^C = \{Pcr_k\}_{k=\overline{0,15}},$$

$$T^D = \{Tda_i\}_{i=\overline{1,17}} \cup \{Tdd_j\}_{j=\overline{1,15}} \cup \{Tdr_k\}_{k=\overline{1,24}}$$

$$T^C = \{Tcr_k\}_{k=\overline{1,15}}$$

$$T_a^D = \{Tda_i\}_{i=\{2,4,6,8,10\}} \cup \{Tda_{12}\}$$

$$tempo(Tda_i)_{i=\{2,4,6,8,10,12\}} = \{9.5, 9.3, 8.5, 0.5, 4.75, 27.2\}$$

$$T_d^D = \{Tdd_r\}_{r=\{1,4,7,10,13\}}$$

$$tempo(Tdd_r)_{r=\{1,4,7,10,13\}} = (d_{dd_r})_{r=\{1,4,7,10,13\}}$$

$$T_r^D = \{Tdr_l\}_{l=\{4,9,14,19\}}$$

$$tempo(Tdr_l)_{l=\{4,9,14,19\}} = (d_{dr_l})_{l=\{4,9,14,19\}}$$

S-a considerat viteza medie de deplasare a RM: $v_r = 94mm/s$.

Modelul SHPN a fost realizat sub platforma VisualObjectNet++ specializată în modelari de procese discrete sau continue (Fig.8, Fig.9) dar cu anumite limitari în ceea ce privește posibilitățile de simulare.

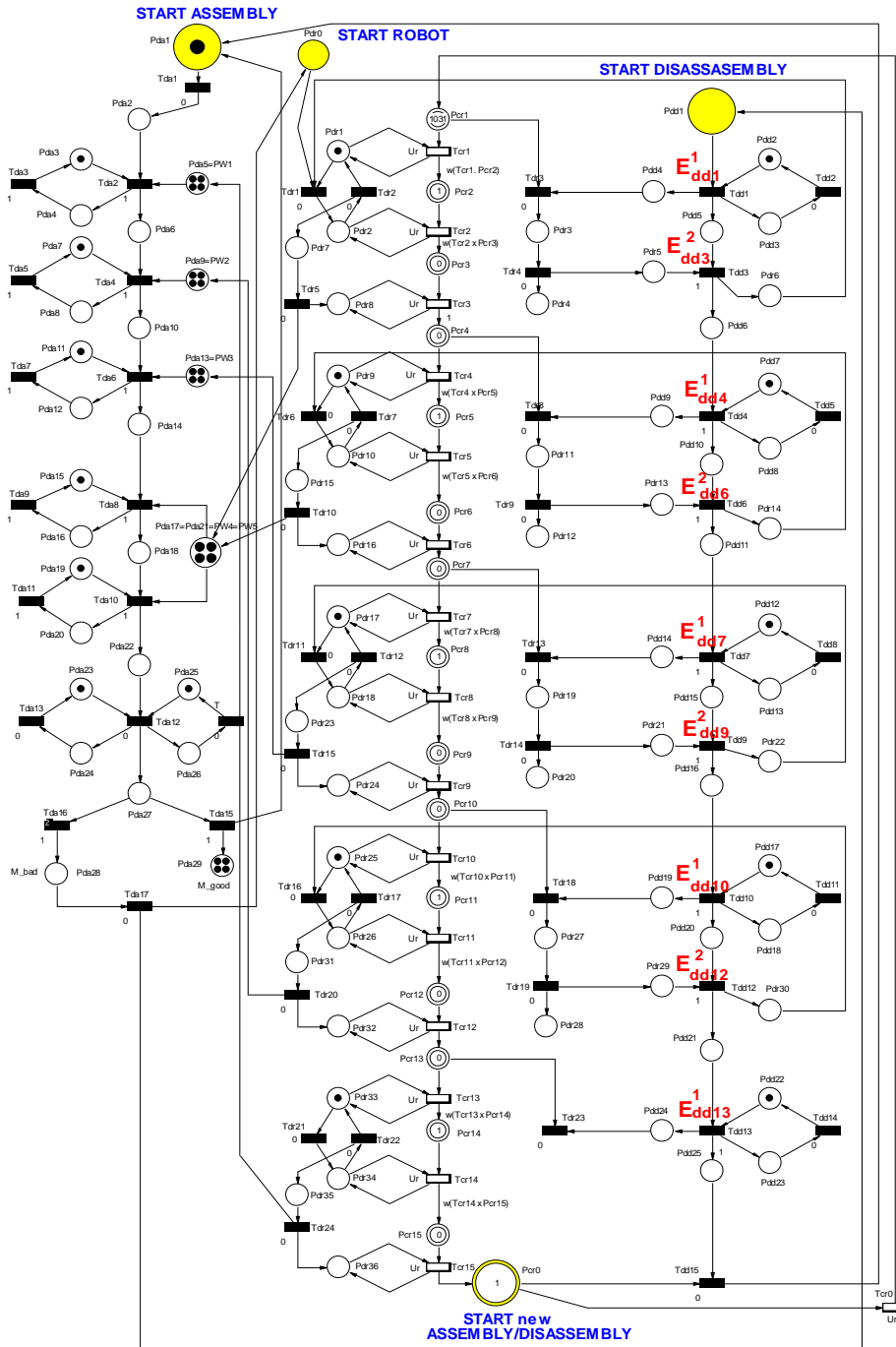


Figura 8. Modelul SHPN corespunzător liniei macatronică HERA & Horstman

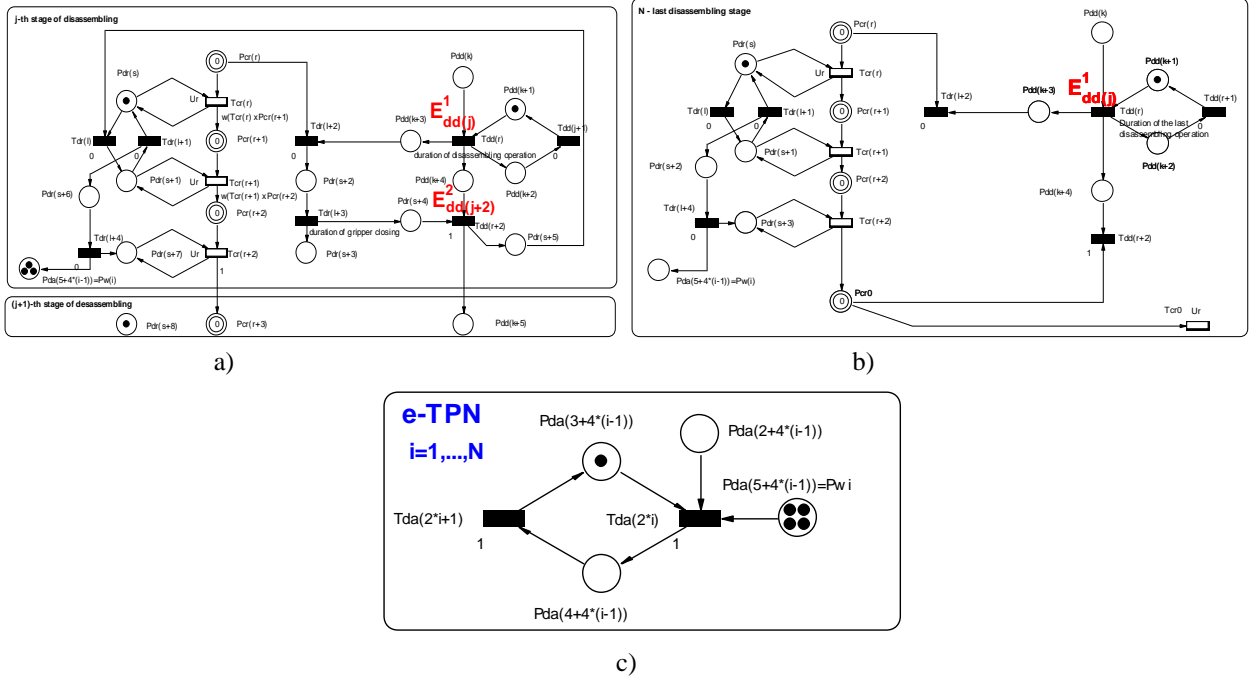


Figura 9. Modele e-SHPN corespunzătoare unei operații elementare: a) de dezasamblare ; b) ultima operație de dezasamblare; c) de asamblare

II.3.4. Rezultate ale simulării off-line a modelului SHPN

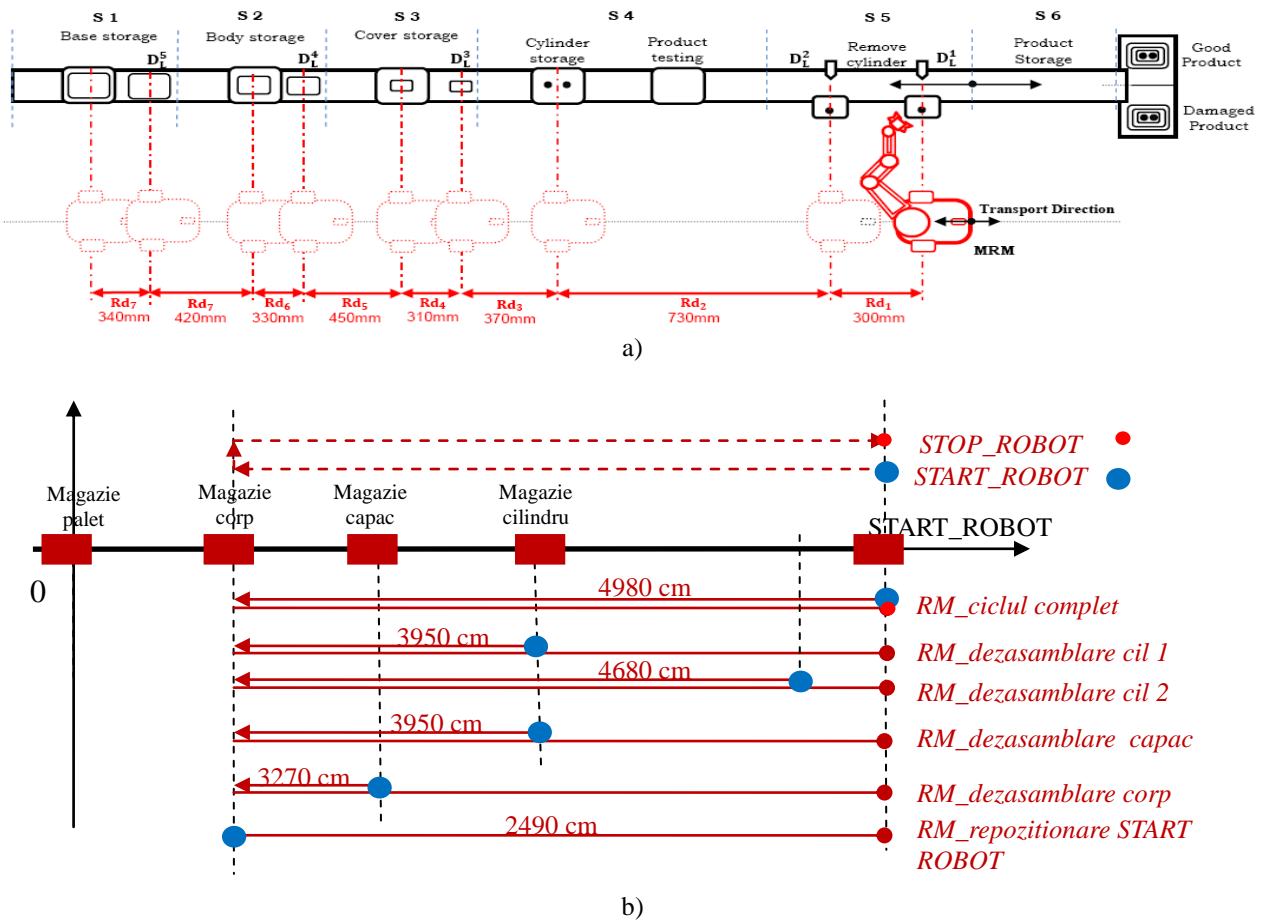


Figura 10. a) Localizarea stațiilor de dezasamblare și a magaziiilor de depozitare; b) Distanțele parcurse de RM în raport cu poziția de STOP (repozitionare în vederea unei noi dezasamblări)

Tabel 1.

<i>Etapă de dezasamblare/depozitare</i>	$M_0(Pcr_{(r)})$ <i>Distanța de parcurs de către RM (până la încheierea dezasamblării)</i>	$M_0(Pcr_{(r+3)})$ <i>Distanța ramasă de parcurs de către RM</i>
I - dezasamblare cilindru 1	4980	3950
II - repositionare RM	3950	4680
III – dezasamblare cilindru 2	4680	3950
IV- dezasamblare capac	3950	3270
V – dezasamblare corp	3270	2490
V I– repositionare robot la START	2490	0

Pentru simularea off-line a modelului SHPN s-a neglijat recepția semnalelor de sincronizare. Datorită complexității modelului, pentru a obține rezultate corecte și detaliate, s-a propus simularea subproceselor reprezentând etape succesive ale procesului de dezasamblare (Fig.11). S-a ales mediul de modelare/simulare *Sirphyco* specializat în modelarea proceselor hibride. Deoarece modelul PN asociat *procesului de asamblare* are o tipologie simplă TPN, considerăm ca simularea acestuia nu ar evidenția aspecte deosebite în ceea ce privește viabilitatea, marginirea, reversibilitatea și în final stabilitatea și lipsa de blocaje a modelului.

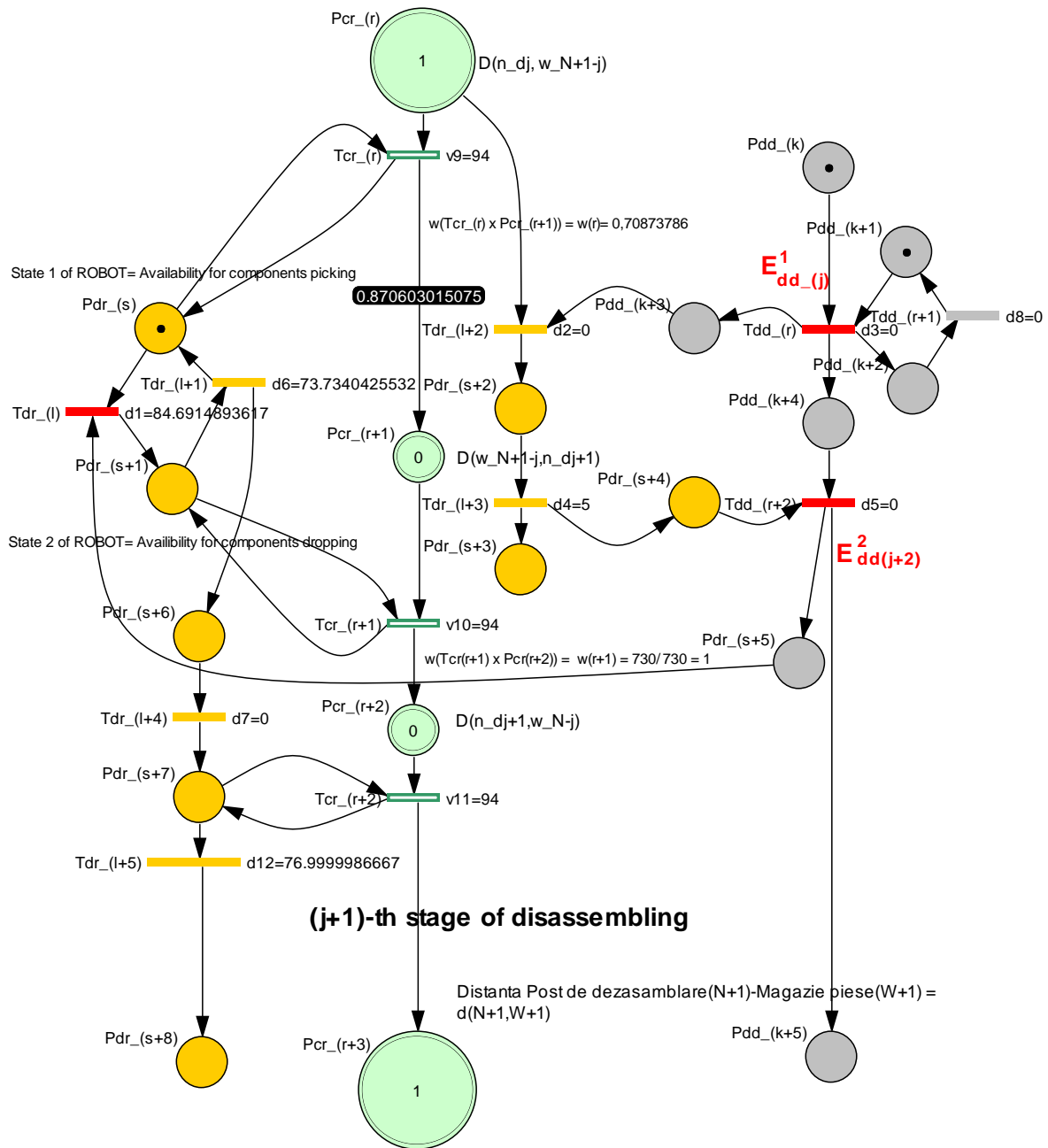
Pentru fiecare etapă din cadrul *procesului de dezasamblare*, am delimitat din modelul general (Fig.8) modulele elementare cu tipologie hibridă. Aspectul hibrid este imprimat de variațiile discrete și continue ale stărilor RM și piesei supusă dezasamblării. Stările cu variație continuă sunt asociate variațiilor continue de stare pentru RM, prin variabila “*distanță ramasă de parcurs*” de către RM în raport cu momentul final al dezasamblării complete. Tabelul 1 descrie această variabilă și valorile corespunzătoare etapelor succesive ale procesului de dezasamblare. Stările discrete asociate RM corespund operațiilor de manipulare/așteptare pe care acesta le execută și sunt modelate cu TPN (rețele Petri temporizate). În mod similar, fiecare operație de dezasamblare este modelată cu TPN. Temporizările asociate tranzițiilor din TPN, corespund duratelor operațiilor de asamblare/dezasamblare precum și duratelor operațiilor de manipulare/așteptare.

Pentru cele șase etape ale procesului complet de dezasamblare, s-au efectuat simulări ale modelelor asociate fiecărei etape. Fiecare model a fost simulat pornind de la un marcaj inițial determinat. Marcajul inițial conține și variabila “*distanță de parcurs*” – $M_0(Pcr_{(r)})$, conform valorilor precizate în Tab.1. și Fig.10. : Fig.12, Fig.15.a, Fig. 17.a, Fig. 19.a, Fig.21a. După simulare, valoarea marcajului $M_0(Pcr_{(r+3)})$ este în acord cu valorile din Tab.1. și Fig.10. : Fig. 13, Fig. 15.b, Fig. 17.b, Fig. 19.b, Fig. 21.b.

Pentru realizarea sincronizărilor considerate în cadrul modelului SHPN, s-au prevăzut temporizări care corespund duratelor în care RM execută operațiile de manipulare (pick-up și dropping)/așteptare precum și secvențe de deplasare în punctele: post de dezasamblare - magazia de depozitare – repositionare la postul următor de dezasamblare. Aceste temporizări sunt asociate tranzițiilor după cum urmează: $tempo(Tdr_{(l)})$, $tempo(Tdr_{(l+1)})$, $tempo(Tdr_{(l+5)})$. Aceste variabile determină pentru RM duratele de manipulare/asteptare/deplasare astfel încât dezasamblarea să se desfășoare ca un proces automatizat și fără întreruperi.

Evoluția stărilor în modelul SHPN rezultă din succesiunea evoluției stărilor în cele șase modele elementare. Toate simulările pun în evidență modele marginite, viabile, accesibile din punct de vedere al oricărui marcaj și fără blocaje. Modelul global SHPN are aceleași proprietăți la care se adaugă și aceea de reversibilitate.

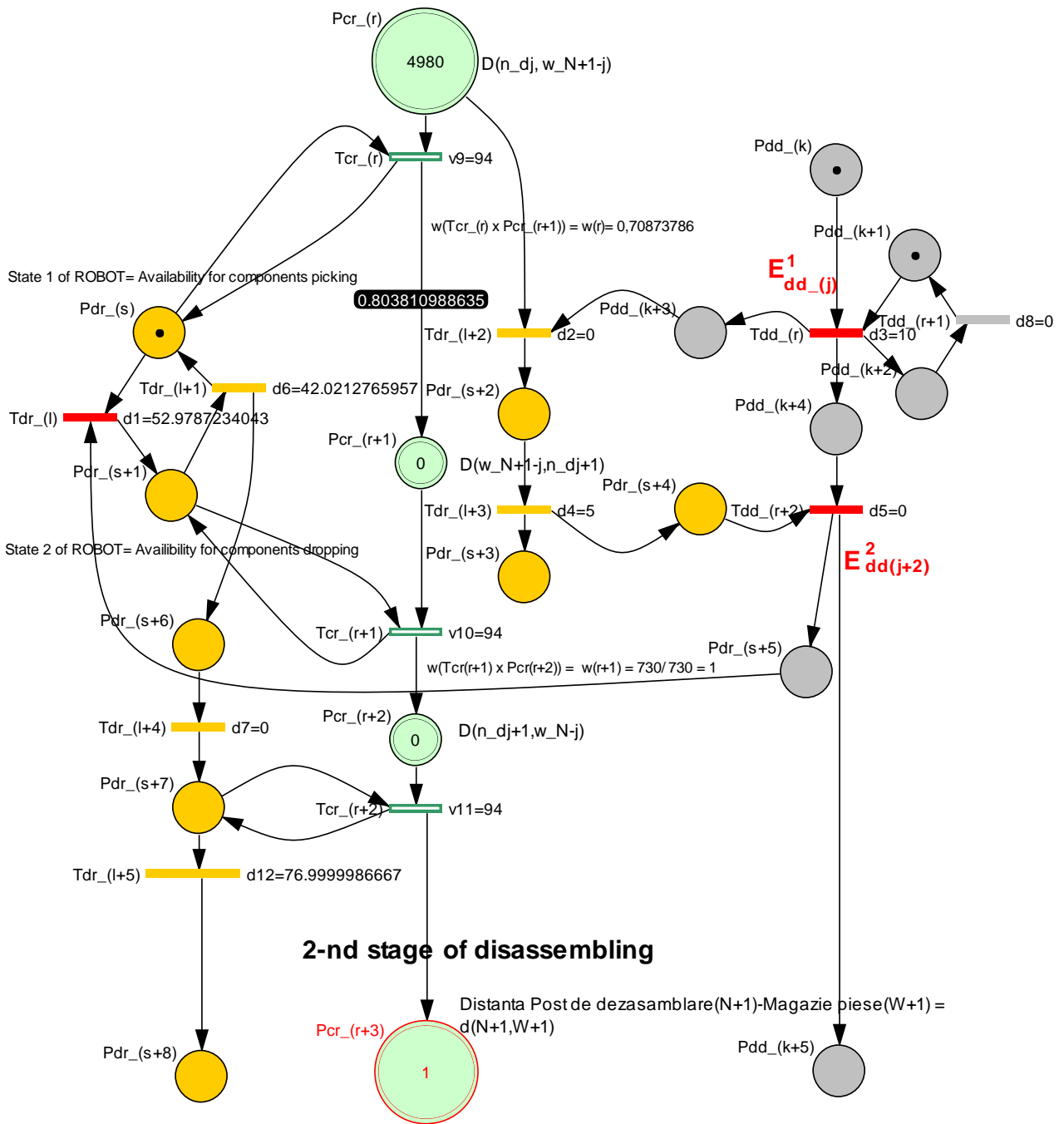
j-th stage of disassembling



(j+1)-th stage of disassembling

Figura 11. SHPN corespunzator unei operații elementare “j” de dezasamblare deservită de un RM

1-st stage of disassembling



1-st stage of disassembling

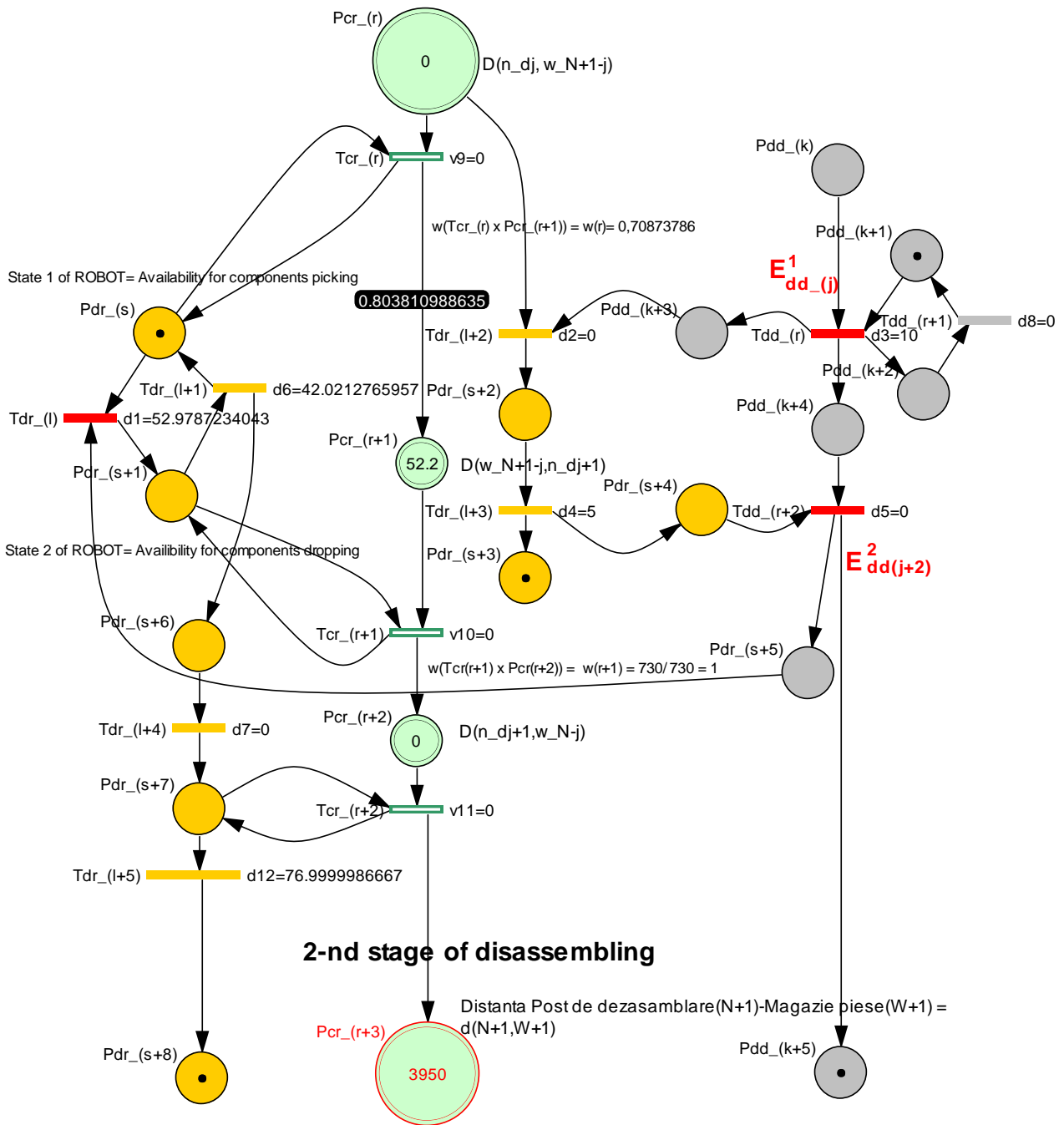


Figura 13. Marcajul final al SHPN corespunzator primei operații elementare de dezasamblare – “dezasamblare cilindrul 1” deservită de un RM

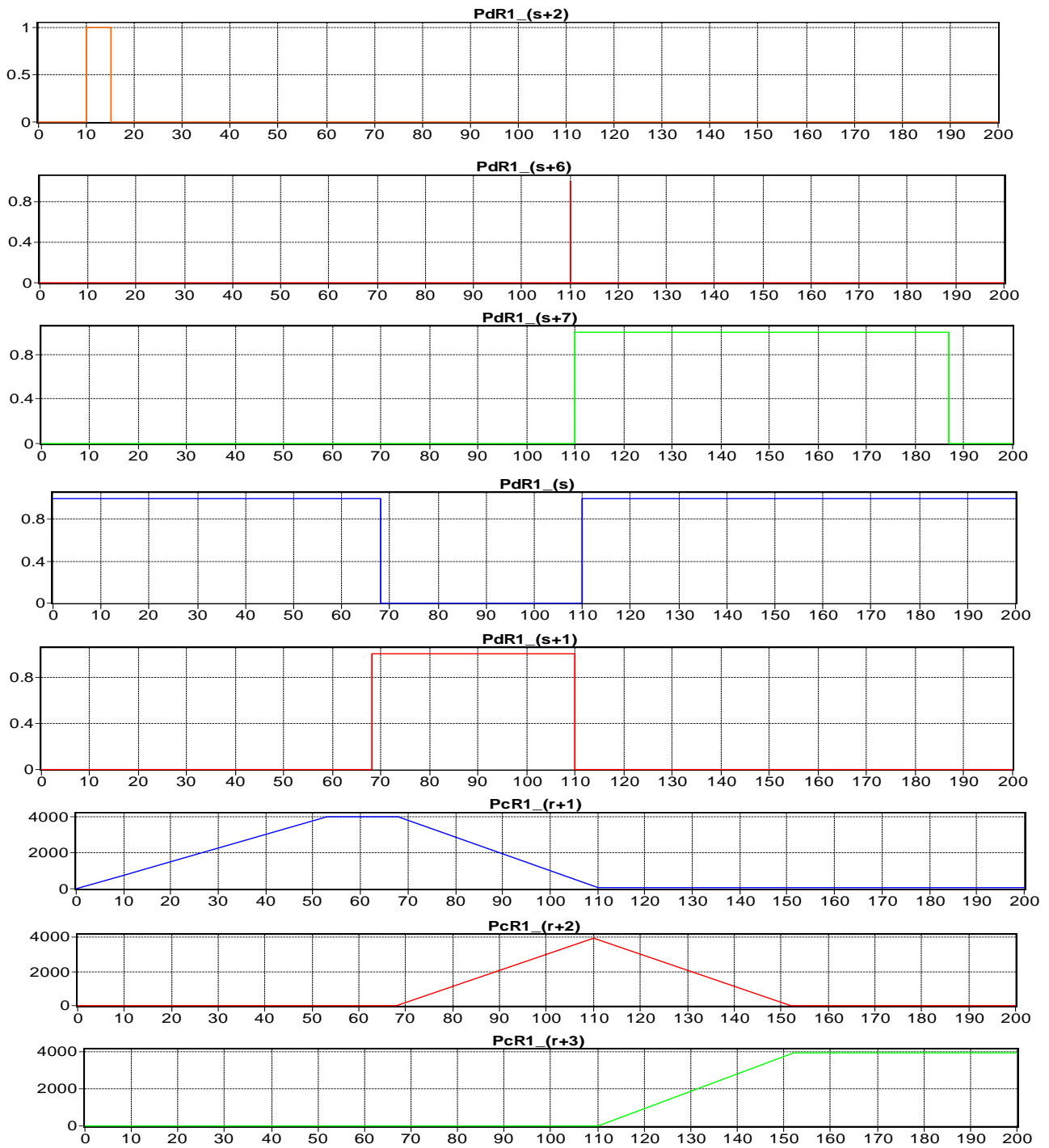


Figura 14. Evolutia starilor în modelul SHPN corespunzator primei dezasamblari

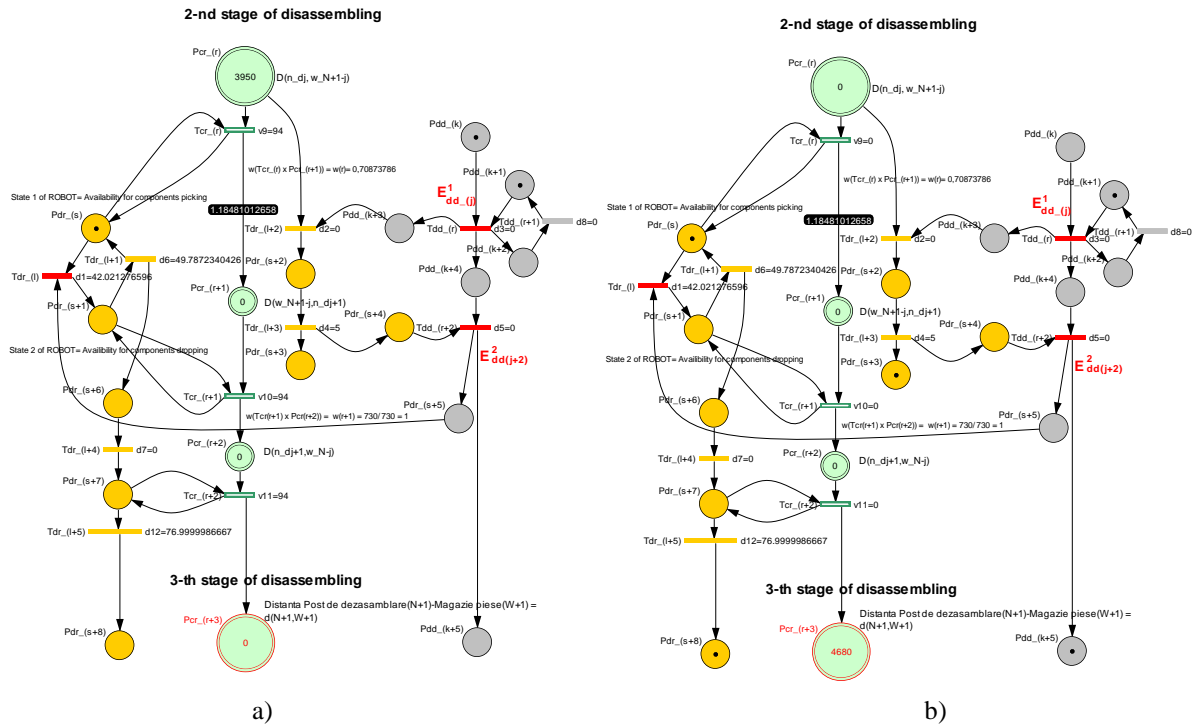


Figura 15. SHPN corespunzator repozitionării RM la postul dezasamblare “cilindrul 2”; a) Marcaj inițial b) Marcaj final

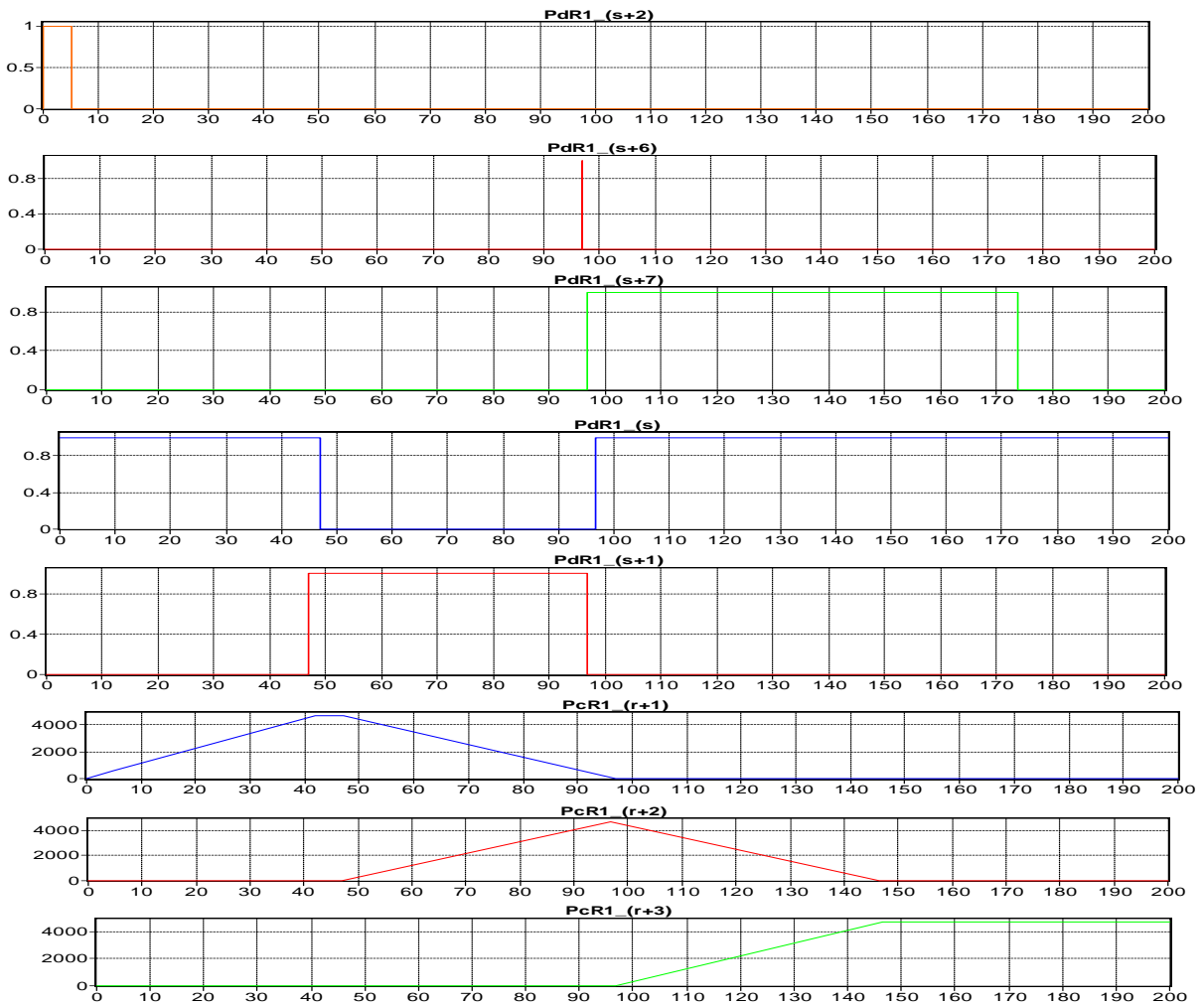


Figura 16. Evolutia stărilor în modelul SHPN corespunzator repozitionării RM la postul dezasamblare “cilindrul 2”

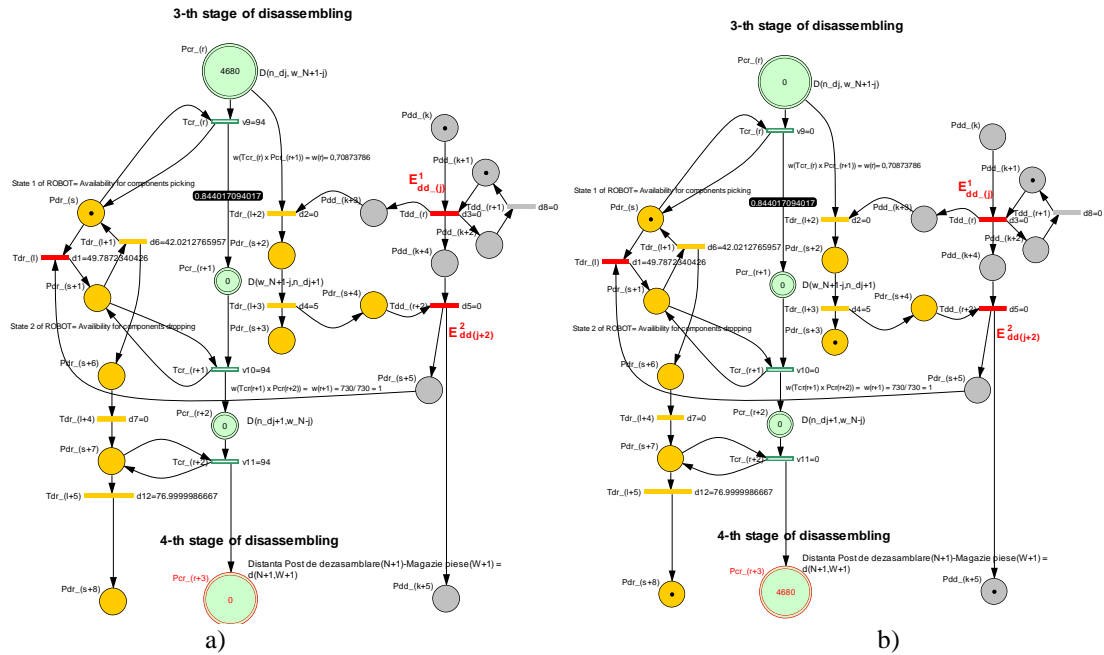


Figura 17. Marcajul inițial (a) și final (b) al SHPN corespunzător operației elementare de dezasamblare – “dezasamblare cilindrul 2” deservită de un RM

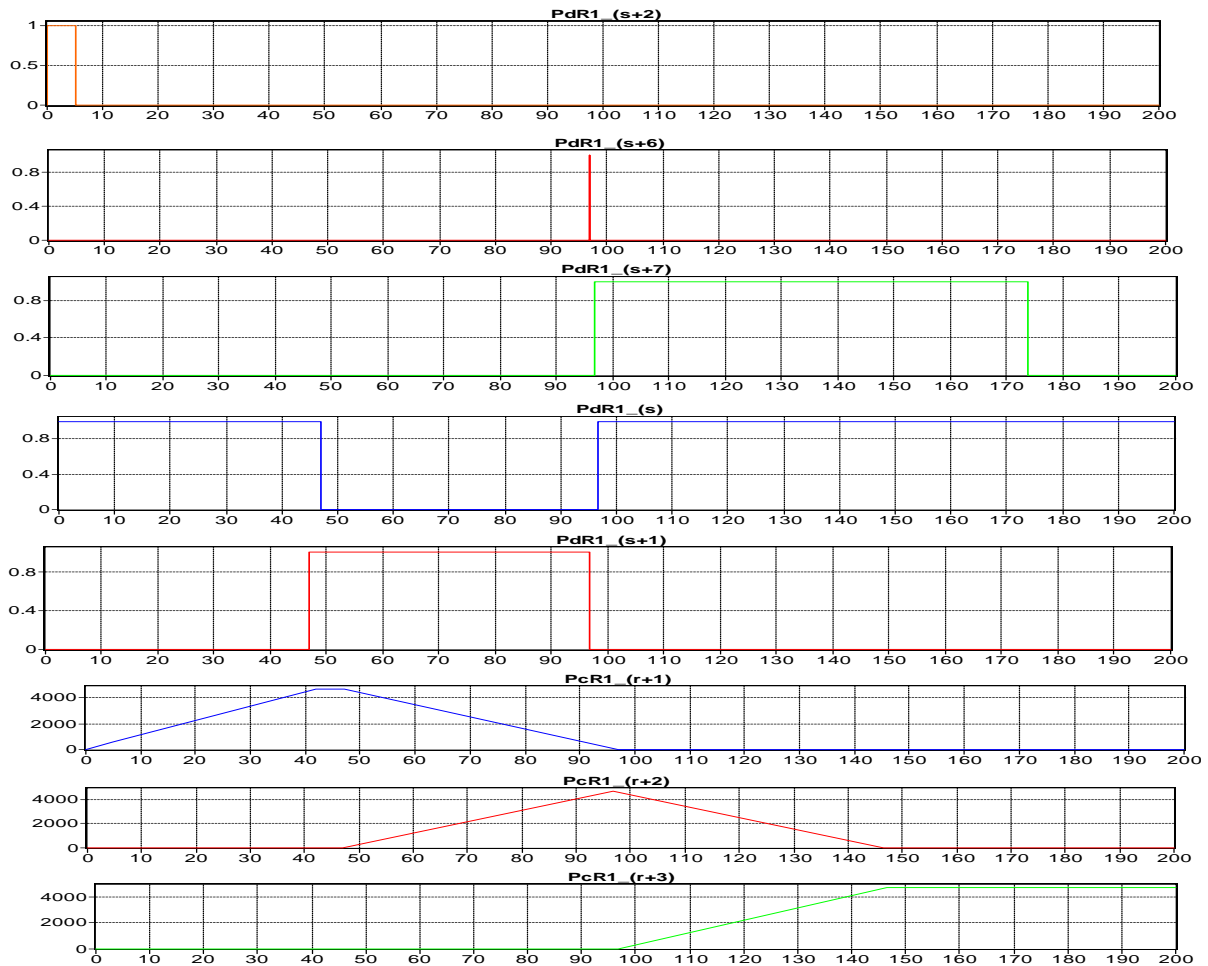


Figura 18. Evoluția stărilor în modelul SHPN corespunzător operației elementare de dezasamblare – “dezasamblare cilindrul 2” deservită de un RM

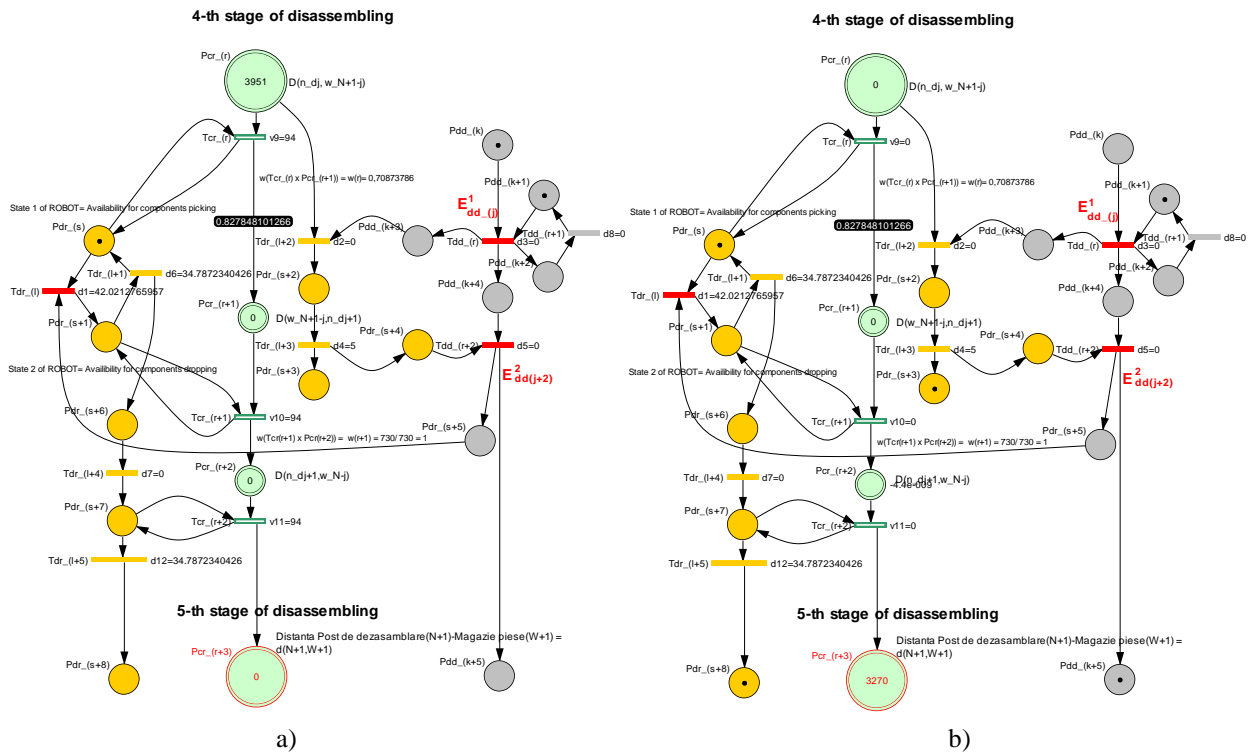


Figura 19. Marcajul inițial (a) și final (b) al SHPN corespunzător operației elementare de dezasamblare – “dezasamblare capac” deservită de un RM

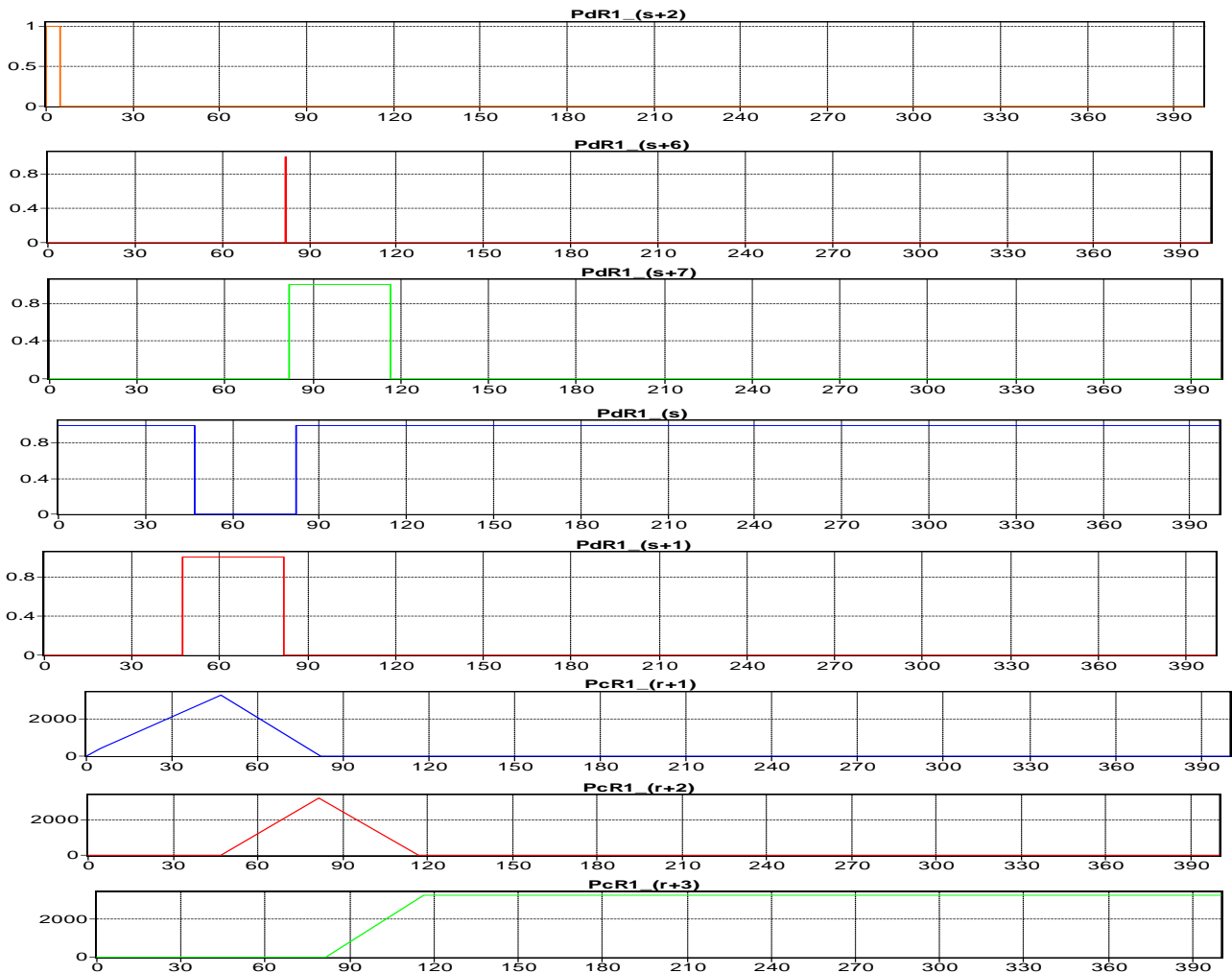


Figura 20. Evoluția stărilor în modelul SHPN corespunzător operației elementare de dezasamblare – “dezasamblare capac” deservită de un RM

dezasamblare – “dezasamblare capac” deservită de un RM

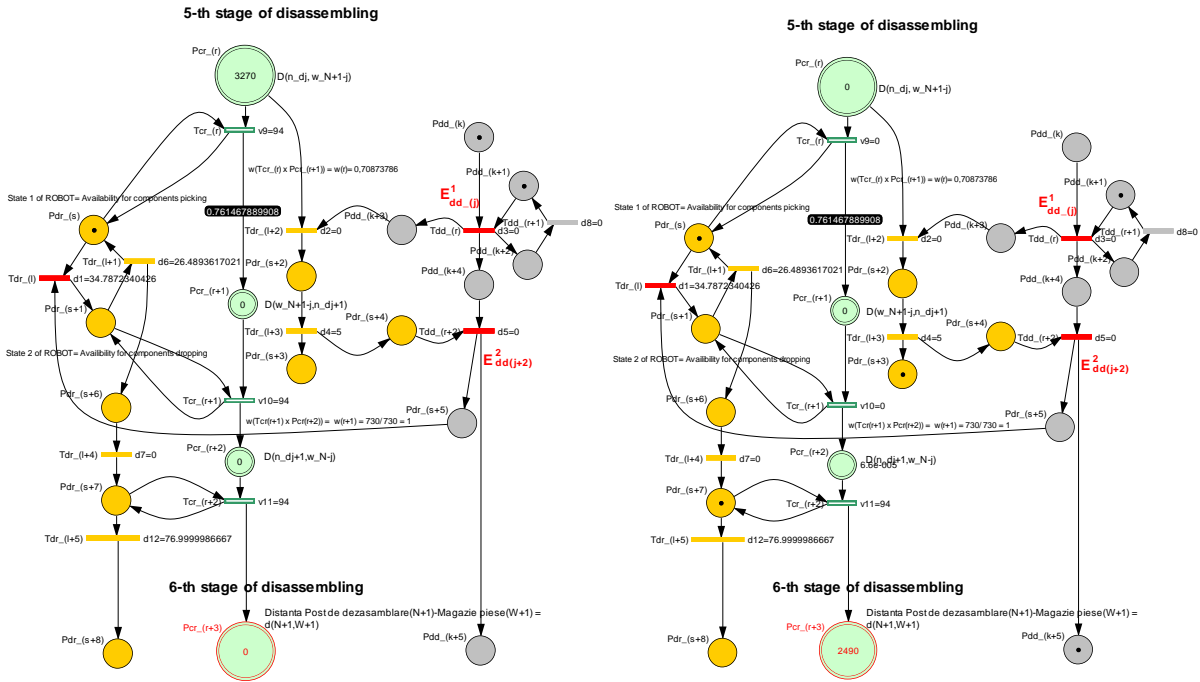


Figura 21. Marcajul inițial (a) și final (b) al SHPN corespunzător operației elementare de dezasamblare – “dezasamblare corp” deservită de un RM

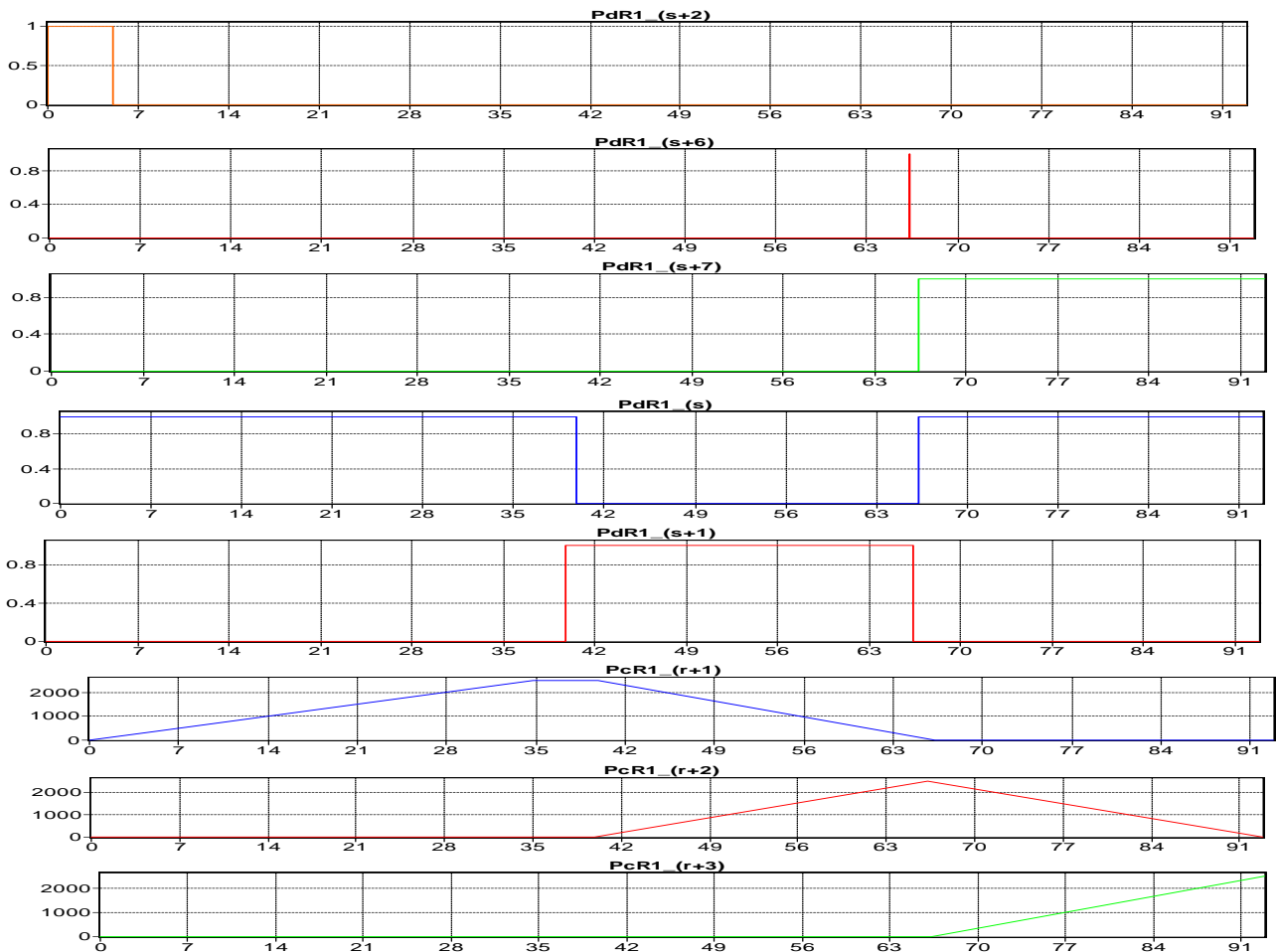


Figura 22. Evoluția stărilor în modelul SHPN corespunzător operației elementare de dezasamblare – “dezasamblare corp” deservită de un RM

II.3.6. Modelarea cu SHPN a unei secvențe de dezasamblare deservită de doi roboți colaborativi

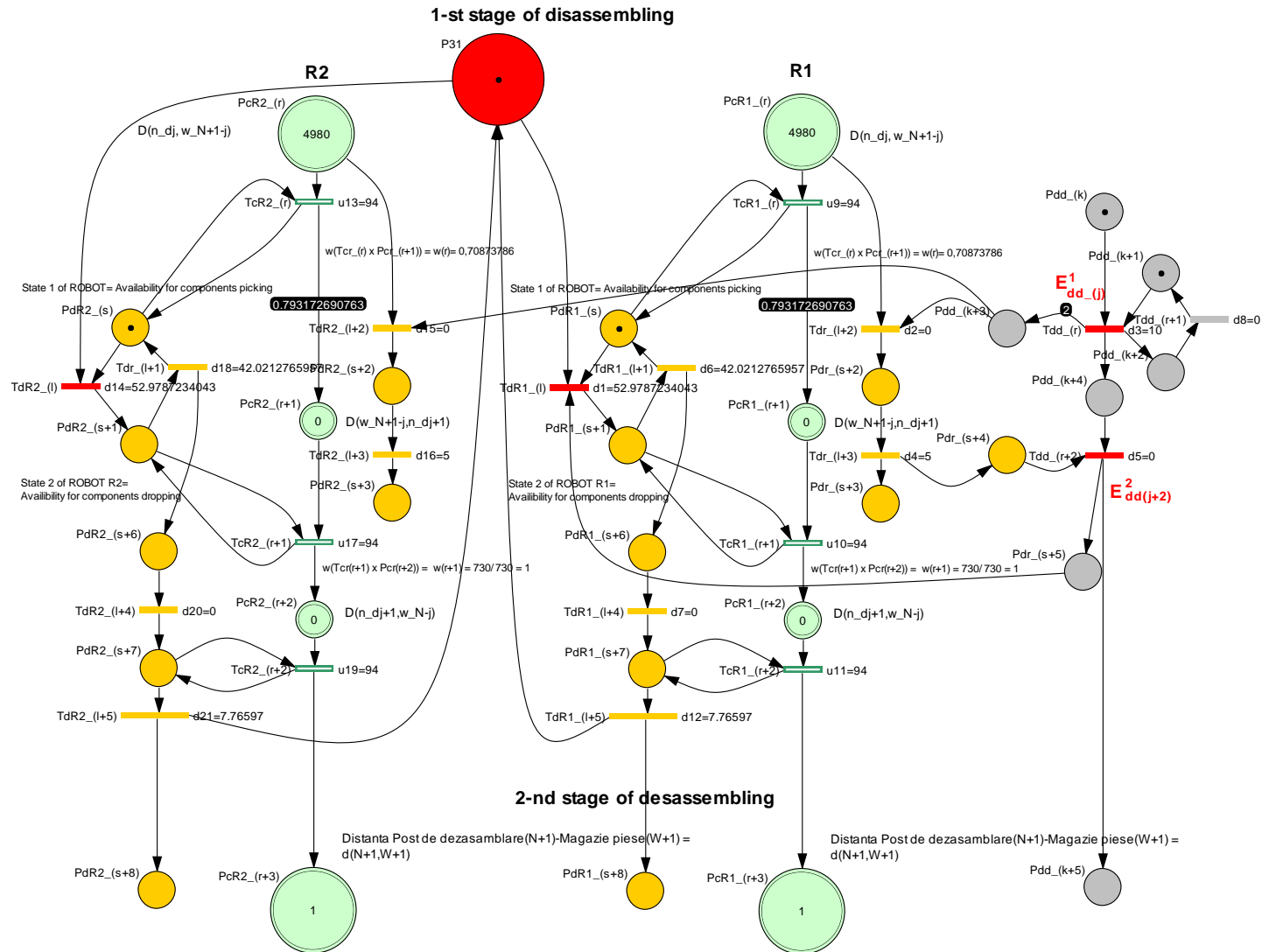


Figura 23. Modelul SHPN corespunzător unei secvențe de dezasamblare deservită roboți colaborativi

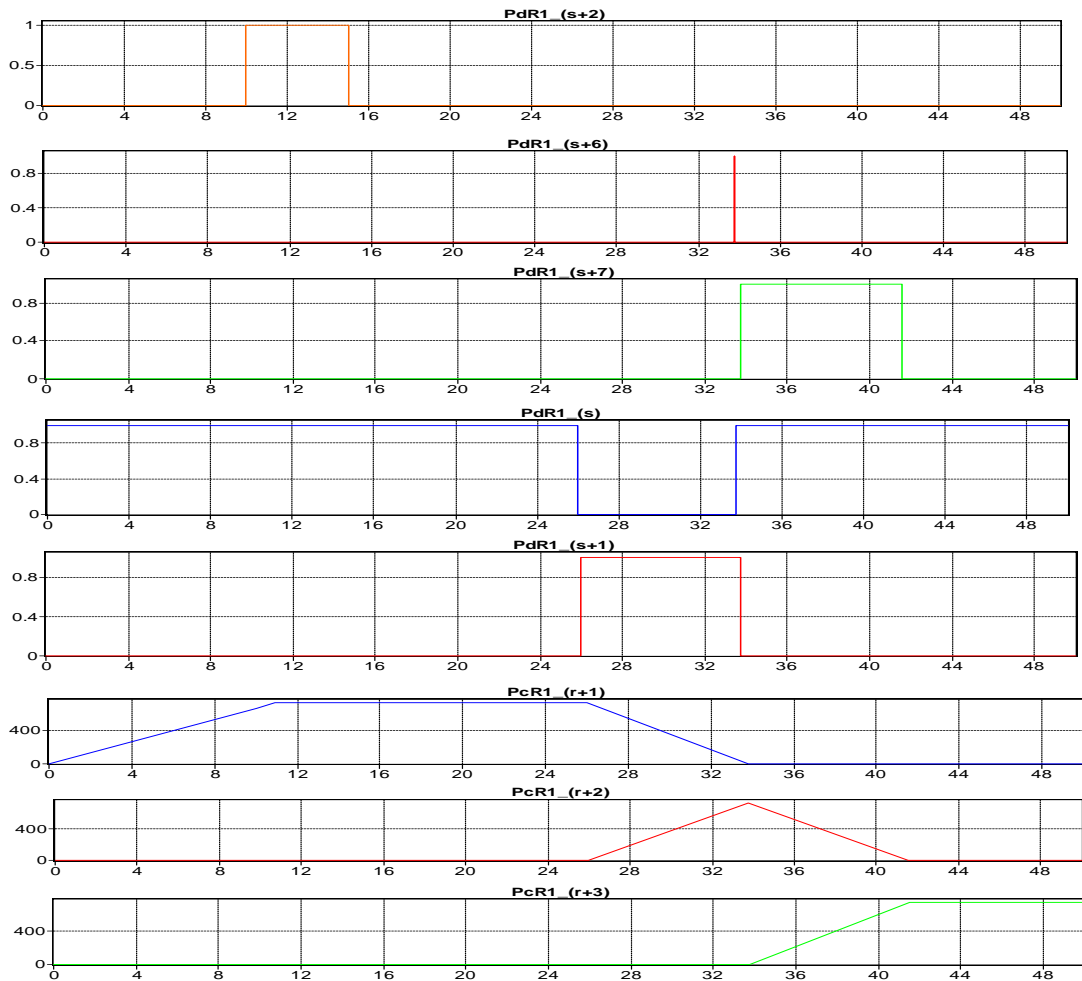


Figura 24. Evoluția stărilor în modelul SHPN corespunzător unei secvențe de dezasamblare deservită roboți colaborativi

În abordările anterioare s-a considerat ca piesa supusă asamblării / dezasamblării este de greutate mică sau medie iar viteza de 94 mm/sec.. În procesul de deservire a dezasamblării s-a utilizat un RM de la Adept Mobile Robots: Pioneer 3DX (Fig.25) care poate transporta greutăți de până la 17 kg cu o viteză maximă de 1.6 m/sec.:

(<http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx>)



Fig. 25. Linia mecatronica HERA & Horstmann deservită de RM- Pioneer 3DX

Pentru dezasamblare /transportul pieselor de greutate mari se pot utiliza RM dedicați care pot deservi linia de dezasamblare, alături de RM destinați manipulării pieselor de greutate mici. Cei doi roboți lucrează fie în regim colaborativ (Fig. 23) fie în paralel - unul destinat manipulării iar celălalt transportului piesei între posturile de lucru (Fig. 26).

Pentru cele două situații s-au realizat modelele SHPN corespunzătoare unei etape elementare

generice de dez asamblare. In ambele cazuri, în evoluția stărilor procesului, nu survin modificări față de cazurile anterioare (Fig.24). In regimul colaborativ cele două resurse de deservire lucrează exclusiv, modelul SHPN având tipologia de sistem cu resurse partajate. Se poate imagina ca cele două sisteme robotice deservesc doua linii paralele de asamblare/dezasamblare. Sistemul de control al celor două platforme robotice prevede pentru fiecare etapa elementara evitarea coliziunilor (Fig.23, P31)

II.3.7. Model SHPN simplificat al unei secvențe de dez asamblare deservită de 2 roboți cu funcționare paralelă: robot echipat cu manipulator și robot destinat transportului greutatilor mari (Pioneer și PatrolBoot)

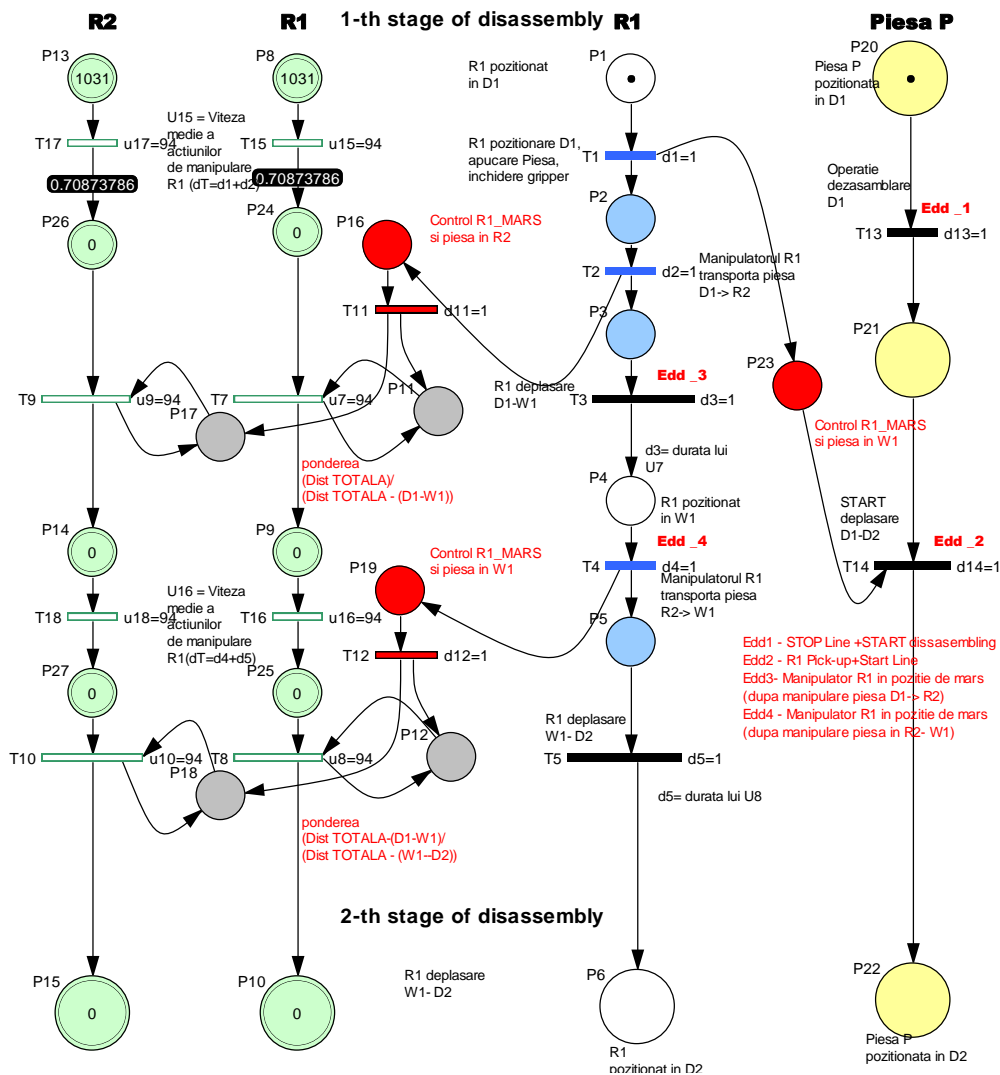


Figura 26. Modelul SHPN al unei secvențe de dez asamblare deservită de doi roboți cu funcționare paralelă

In cazul utilizării celor doi RM cu funcționare în paralel, se vor utiliza patru semnale de sincronizare destinate validării operațiilor de încărcare/descărcare a RM destinat transportului piesei de greutate mare (Fig.26):

- Edd1 - STOP Line + START dez asamblare
- Edd2 - R1 Pick-up + Start Line
- Edd3- Manipulator R1 în pozitie de MARȘ (dupa manipulare piesa D1-> R2)
- Edd4 - Manipulator R1 în pozitie de MARȘ (dupa manipulare piesa în R2- W1),

precum și două variabile de control: P16 (Control Manipulator_R1_ MARȘ și piesa în R2) și P19 (Control R1_ MARȘ și piesa în W1).

II.3.8. Sistemul mecatronic FESTO MPS-200

FESTO MPS-200 reprezintă o linie flexibilă didactică de prelucrare, sortare și depozitare. Aceasta este compusă din 4 stații (celule), fiecare realizând operații diferite.



Figura 27. Sistem flexibil de fabricație FESTO MPS 200

Structura sistemului flexibil FESTO MPS-200 este alcătuită din următoarele stații:

- celula de sortare (Fig.27.a) preia componenta prelucrată de la stația anterioară cu ajutorul unui manipulator axial echipat cu gripper pneumatic, după ce a fost supusă unui test de culoare. Dacă culoarea diferă față de cea corectă, sistemul pneumatic preia piesa și o depune într-o magazie alăturată. Magazia de componente este compusă din două părți. Fiecarei părți i se atribuie o piesă de o anumită culoare. Dacă piesa a trecut testul de culoare, aceasta nu va mai fi stocată în magazie, sistemul de manipulare axial o depune într-o locație a stației următoare.

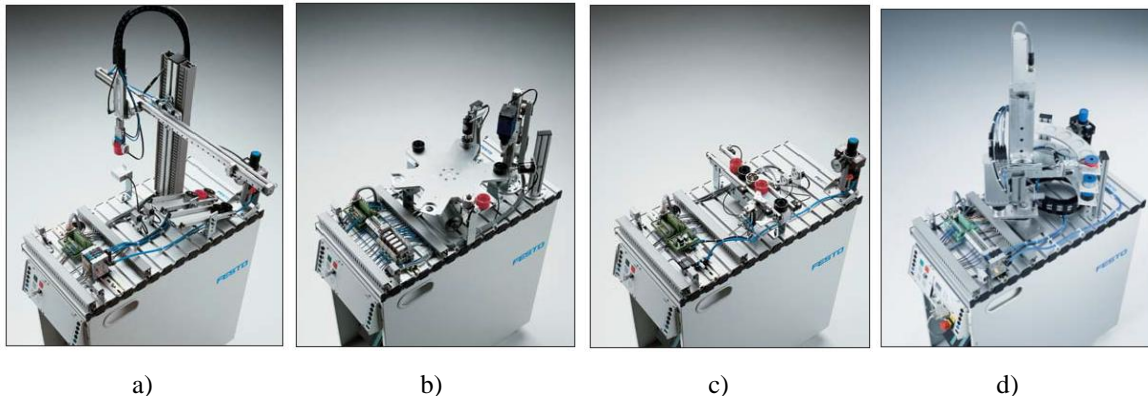


Figura 28. a) Stație de sortare; b) Stație de prelucrare; c) Stație de acumulare piese asamblate; d) Stație de depozitare

- celulă de prelucrare (Fig.27.b) execută două operații distincte, una de găurire și cealaltă de șlefuire a componentei provenite de la stația de testare și sortare. Celula de prelucrare prezintă un sistem de acumulare de tip masă rotativă a șase piese. Prin fiecare rotație a sistemului rotativ se poziționează câte două piese simultan pentru a fi prelucrate. Operațiile de găurire și alezare se execută simultan de către 2 mașini-unelte. După terminarea ambelor prelucrări a unei piese sistemul de acumulare rotativ se poziționează în dreptul următoarei stații. Celula de prelucrare este echipată cu un sistem de senzori de proximitate pentru o poziționare precisă în dreptul celor două mașini-unelte.

- celulă de acumulare (de tip buffer) (Fig.27.c): are rolul de a prelua și stoca piesele provenite de la stația de asamblare. Poate stoca un număr de 5 piese, care vor fi trimise individual către stația

următoare la anumite intervale de timp. Această celulă este echipată cu un sistem de senzori pentru monitorizarea a nivelului de încărcare, un sistem pneumatic de oprire/eliberare piese.

- celulă de depozitare (Fig.27.d): în urma procesului de sortare sunt selectate ordinea și locația unde vor fi depozitate produsele finale. Sortarea produselor este realizată cu ajutorul unui sensor de culoare. Sistemul de depozitare este compus din 3 magazine, fiecărei magazii stochează produse de o anumită culoare. Depozitarea produselor se face cu ajutorul unui manipulator axial echipat cu un gripper pneumatic, care preia produsul de pe banda transportoare a stației anterioare și o depune în depozit. Manipulatorul axial este controlat cu ajutorul unui controller MTR-DCI-42S. Celula de depozitare reprezintă ultima stație din cadrul sistemului flexibil de fabricație FESTO MPS-200 având o capacitate de stocare de 18 produse prelucrate.

II.3.9. Modelul HPN corespunzător liniei de reprelucrare FESTO

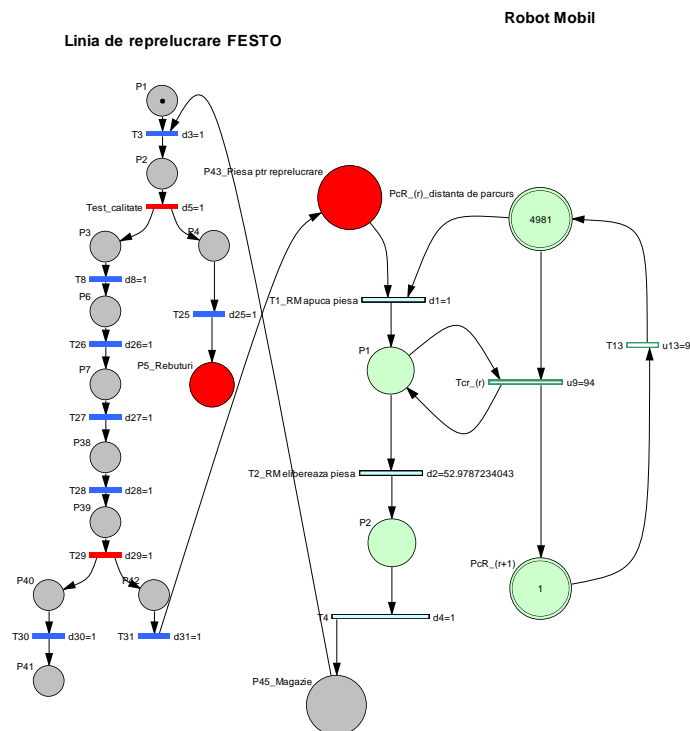


Figura. 29. Modelul HPN corespunzător liniei mecatronice

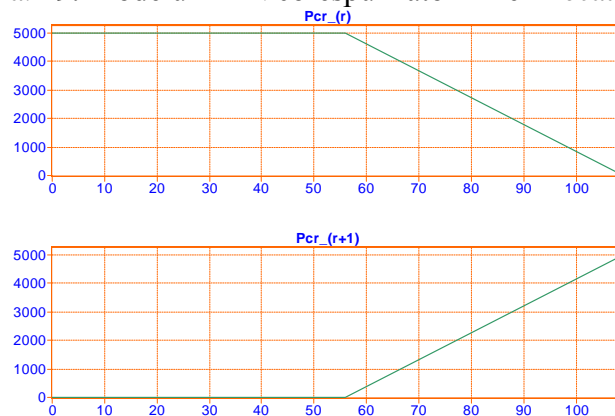


Figura. 30. Evoluția stării RM în timpul deplasării/transportului piesei către punctul de start reprelucrare

Deoarece RM asistă linia de procesare/reprocesare doar în operațiile de manipulare/transport a piesei rebut pentru o nouă reprelucrare, considerăm ca modelul hibrid nu aduce informații calitative semnificative după simularea modelului HPN, relativ la viabilitatea, reversibilitatea și eventualele

blocaje ale procesului real.

Activitatea II.4: Proiectarea structurii de conducere sliding-mode si backstepping, pentru urmarirea unei traiectorii impuse bazata pe modelul cinematic al Prototipului 3. Proiectarea sistemului de navigatie a Prototipului 3, bazata pe sistem laser.

INTRODUCERE: Unitatile autonome formate din platforme mobile echipate cu manipuloarerooboticesuntcerute in domeniindustriale variate intrucat prezinta avantajeprecum cresterea productivitatii, reducerea pericolelor tehnologice asupra factorului uman, reducerea costurilor de productie, etc care pot fi valorificate pentru automatizarea si robotizarea unor procese de productie complexe si periculoase (industrii stiintifice si a ceramicii, metalurgica). Platformele mobile omnidirectionale au capacitatea de a se deplasa in orice directie indiferent de pozitia curenta. Suprafetele de deplasare trebuie sa fie plane, teren neaccidentat (platforme betonate, metalice, vopsite sau nu), fara rugozitati, crapaturi. Platforma se va utiliza in mediul industrial, interior sau exterior.

Platforma mobila are doua componente principale:

- Platforma mobila
- Brat robotic 6-DOF

PLATFORMA MOBILA: Platforma mobila este un prototip proiectat si executat de Electro-Total este o constructie sudata realizata din materiale metalice uzuale (tevi, profile, tabla – OL35, OLT35) cu masa totala de 500 kg. Impreuna cu bratul robotic sicelelalte componente descrise mai jos platforma mobila echipata cu bratul robotic cantareste 920 kg. Structura platformei are in vedere atat realizarea mobilitatii si asamblului platforma-brat robotic cat si operarea in siguranta a bratului robotic pentru procesele de productie propriu-zise, in cazul aplicatiei curente, pentru extragerea de sticlă topită dintr-un cuptor de mentinere si plasarea acesteia intr-o presă pentru realizarea producției de serie mare.

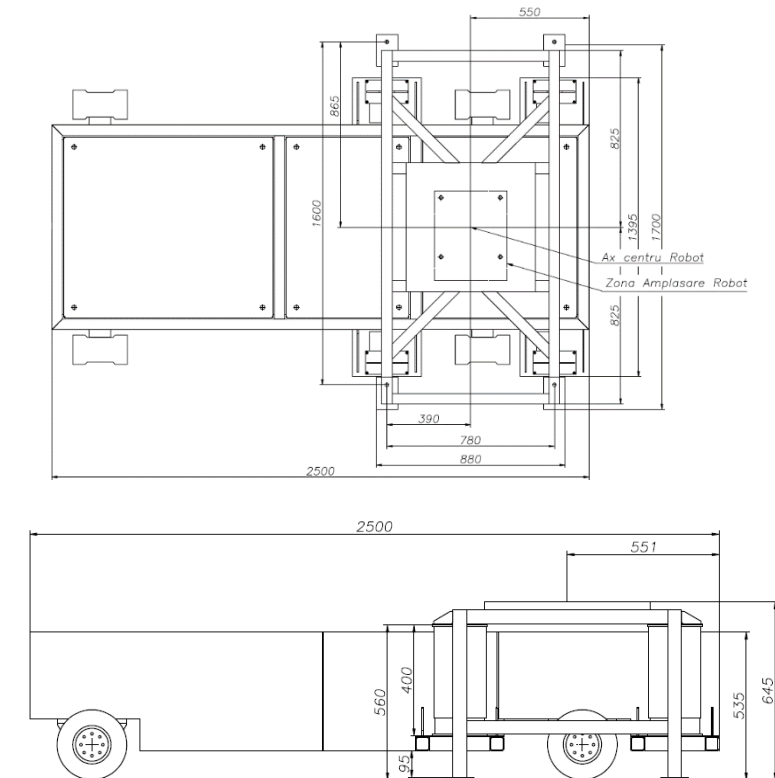


Componentele platformei (in afara de constructia sudata):

1. Roti tip mecanum atasate la platforma cu ajutorul lagarelor cu rulmenti
2. Servosisteme format din servomotor si inverter pentru fiecare roata mecanum
3. Sistem de conducere care implementeaza cinematica rotilor mecanum, memoreaza traseul, coordoneaza navigatia
4. Sursa autonoma de energie electrica

Dimensiunile generale sunt urmatoarele:

- Lungimea totala: 2500mm
- Latimea in zona cadrului de sustinere a bratului robotic: 1700mm
- Inaltimea pana la placa de montaj brat robotic: 645mm
- Inaltimea cu bratul robotic in pozitia retrasa: 2005mm



1. Platforma mobilă este echipată cu **roți mecanum** inventate de către Bengt Ilon de la compania Mecanum AB, Suedia, în 1973. Această roată este compusă dintr-o zonă/disc central (hub) la care sunt atașate un număr de role (roller) la un anumit unghi în jurul discului central. Rolele periferice înclinat transferă o parte din mișcarea aplicată pe direcția de rotație în mișcarea de translație forțată a roții să execute o mișcare perpendiculară pe axul roții. [1] [2].

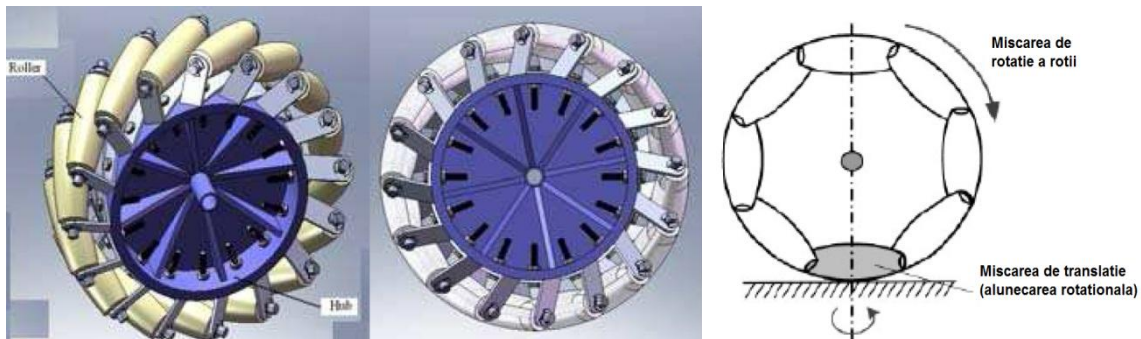


Figura. 1

În funcție de sensul de rotație și viteza fiecărei roți, combinația rezultată a tuturor acestor forțe produce un vector forță total în orice direcție dorită, permițând astfel platformei să se deplaseze liber în orice direcție fără a schimba direcția roții. Figura 1 prezintă schita roții mecanum inventate de Ilon cu rolele periferice montate la 45° cu pantă spre exterior. Profilul roții este în așa fel încât roata mecanum văzută din față este circulară [3].

O platformă mobilă cu patru roți mecanum va avea posibilitatea mișcării omnidirecționale. Alunecarea este o problemă comună la roata mecanum deoarece are doar o rolă cu un singur punct de contact cu solul la un moment dat. Având în vedere dinamica roții mecanum, se poate crea vectori de forță atât pe direcția x și y în timp ce este acționată doar pe direcția y . Poziționarea a patru roți mecanum, una la fiecare colț al șasiului (două perechi în oglindă), permite compunerea forțelor pe direcțiile x , y . (fig. 2). Rotile mecanum sunt acționate independent de servomotoare montate pe șasiul platformei.

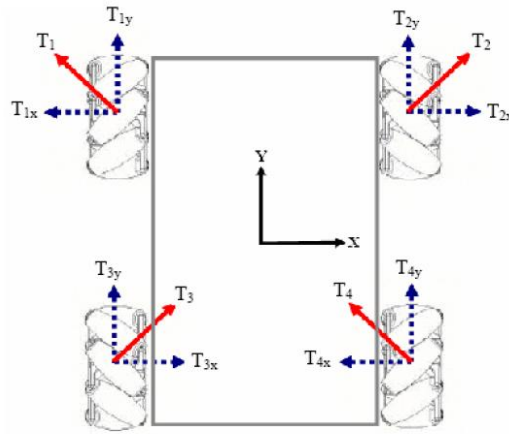
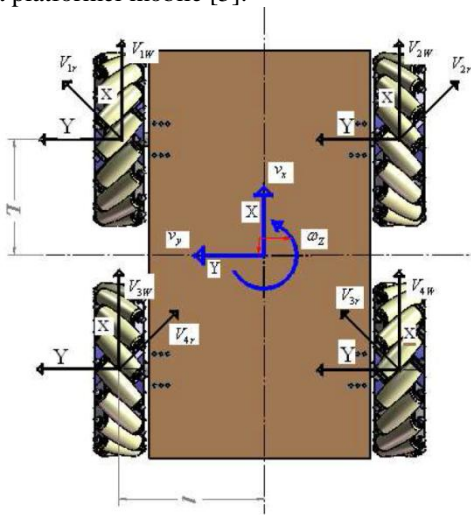


Figura 2. Vectoriforta la roatamecanum

Proiectareaconduceriiplatformeiomnidirectionale are la bazacinematicarotiimecanumreluata succinct maijos. Fortaeactioneazaasuparotiimecanum poate fi descompusa in doua forte component (fig.2) astfel: o componentaeste in directia de rotatiesicealaltapedirectiaaxuluirotii (directiarolelor). Aceasta ultima componentatransferaforta de rotatiecatre role care se rotesc. Princumulareafortelornete ale celorpatru roti ale platformei se determinadirectiagenerala a platformei mobile [5].



$V_{iw} (i = 1,2,3,4) \in R$
 reprezintavectorulvitezacorespunzatordirectiei de rotire a rotii.

$V_{iw} = R_w * \omega_{iw}$
 Unde, in ecuatiavitezeiintervinerazasiviteza unghiulara a rotii.

Figura 3

$V_{ir} (i = 1,2,3,4) \in R$
 Reprezintavectorulviteza tangential ale rolelor care se rotesc liber sisunt in contact cu solul.

V_{iX} este compus din vitezele V_{iw} .

Similar:

$$V_{iY} = V_{ir} / \cos 45$$

$$V_{iX} = V_{iw} + (V_{ir} / \sqrt{2}), V_{iY} = V_{ir} / \sqrt{2} \quad (1)$$

$$V_{iX} = v_x - l\omega_z, V_{iY} = v_y + L\omega_z \quad (2)$$

unde $v_x, v_y, \omega_z \in R$ reprezinta descompunerile vitezei si vitezei unghiulare pe axele x si y ale platformei. L este distanta pe axa X dintre axa fiecarei roti si centrul de greutate al platformei; l este distanta pe axa Y dintre fiecare roate si centrul de greutate.

Din ecuatiile (1) si (2) se obtin:

$$V_{1W} = v_x - v_y - (L + l)\omega_z \quad (3)$$

$$V_{2W} = v_x - v_y - (L + l)\omega_z \quad (4)$$

$$V_{3W} = v_x - v_y - (L + l)\omega_z \quad (5)$$

$$V_{4W} = v_x - v_y - (L + l)\omega_z \quad (6)$$

Ecuatiile (3)-(6) se introduc in ecuatia de cinematicainversa:

$$V_W = J V_0 \quad (7)$$

unde $V_0 = [v_x, v_y, \omega_z]^T \in R^3$ este vectorul viteza in coordonate Carteziene iar

$V_W = [V_{1w}, V_{2w}, V_{3w}, V_{4w}]^T \in R^4$ este vectorul unghiular de viteza al rotilor.

R_w este raza rotii.

Punandconditia $\bar{\omega} = 0$ rezulta urmatoareaecuatiede avans:

$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ L+l & L+l & L+l & L+l \end{bmatrix} \begin{bmatrix} R_w \theta_1 \\ \vdots \\ R_w \theta_4 \end{bmatrix} \quad (8)$$

Astfel, in coordonatele (X, Y, Z), viteza platformei mobile rezulta astfel:

$$v_x = \frac{R_w}{4} (\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4) \quad (9)$$

$$v_y = \frac{R_w}{4} (-\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 - \dot{\theta}_4) \quad (10)$$

$$\omega_z = \frac{R_w}{4(L+l)} (-\dot{\theta}_1 + \dot{\theta}_2 - \dot{\theta}_3 + \dot{\theta}_4) \quad (11)$$

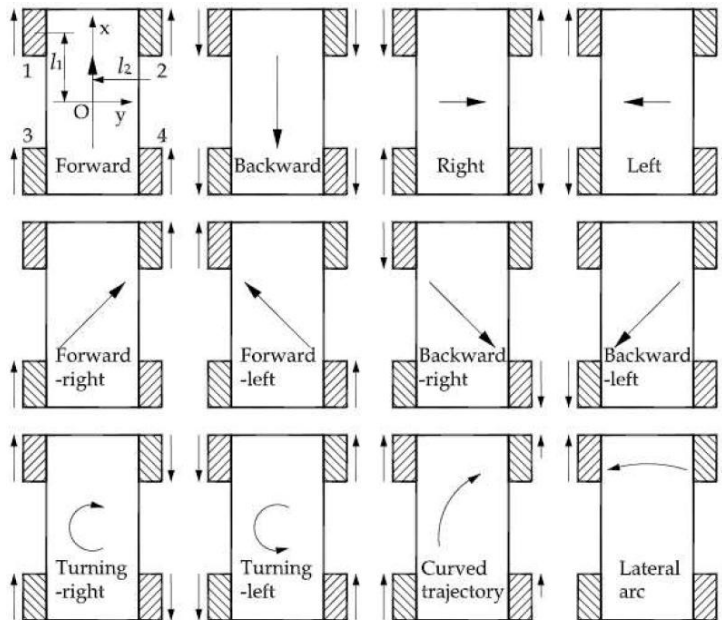
Directia miscarii rezultante β in coordonatele (x,y,z) rezulta astfel:

$$\beta = \tan^{-1} \left(\frac{v_y}{v_x} \right) \quad (12)$$

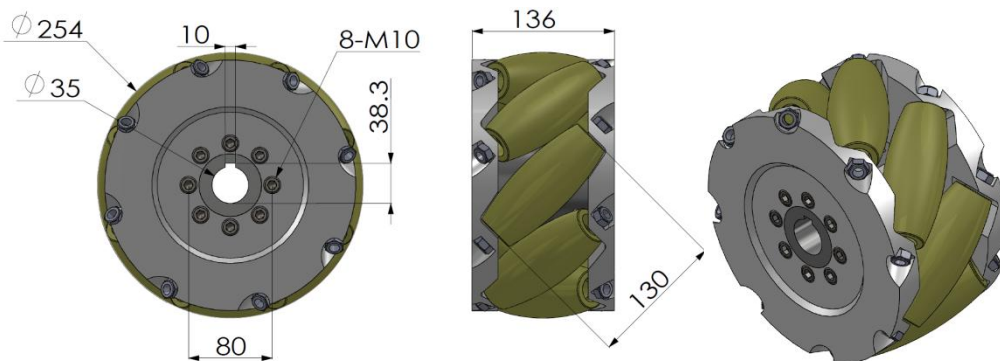
$$v_r = \sqrt{(v_x^2 + v_y^2)} \quad (13)$$

Din ecuatiile cinematice de mai sus putem determina pozitia, directia de miscare si rezultanta vitezei aferente platformei omnidirectionale echipata cu roti mecanum. Ecuatiile de cinematica inversa permit calcularea vitezelor unghiulare pentru fiecare roata pentru a impinge platforma in viteza si rotire dorita. Ecuatiile cinematice de avans permit calcularea miscarii platformei plecand de la vitezele unghiulare aferente celor patru roti.

Directia si rotirea rotilor diagonale sunt controlate independent. Folosind aceasta situatie a rotilor in acelasi timp in timpul operarii platformei se obtin opt directii de deplasare fara a schimba orientarea platformei. Miscarea intre 0° si 360° se obtine prin schimbarea vitezei rotilor diagonale prin controlul tensiunii de alimentare ale servomotoarelor.



Rotile mecanum folosite la platforma mobila au urmatoarele dimensiuni si caracteristici:

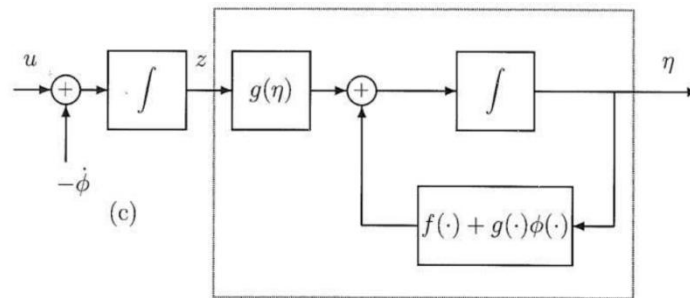
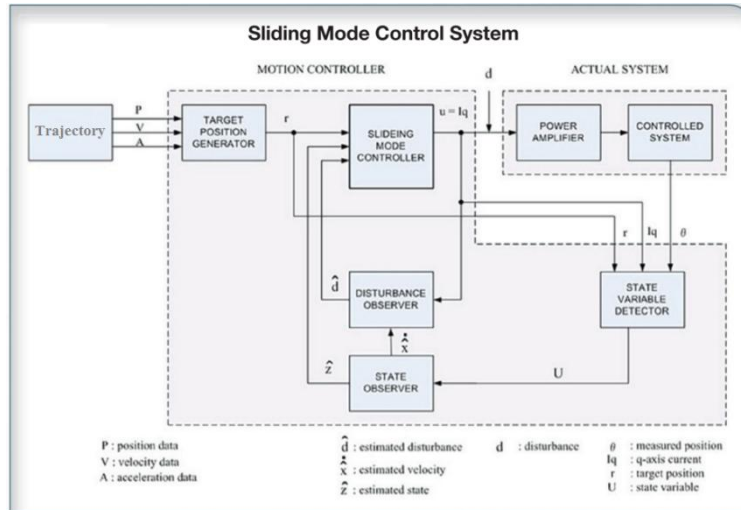


2. **Servosistemele** care antreneaza rotile mecanum utilizate sunt formate din urmatoarele componente:

- servomotoare tip AC M2n, Parker Automation cu rezolver integrat
- echipata cu reductor planetar PGNLM ceofera un cuplu de 110Nm la iesirea din reductor
- Variator Digital 638A alimentat la 230Vca

3. **Conducerea la distanta in navigatia platformei** se realizeaza cu un sistem bazat pe un Automat Programabil / PC

industrial /Controller care implementeaza in structurasoftware / hardware memorarea traiectoriei, conducerea platformei de-a lungul acesteia traiectorii in contextul existentei obstacolelor petraseusi, de asemenea tinand cont de alunecarea specificarotilormecanum (contactul cu solul se face intr-un singur punct la fiecare roata). Astfel, pentru controlul platformei in conditiireale (in prezent incertitudinilor si a perturbatiilor) se studiazaeficienta in vederea implementarii mai multor metode precum **sliding mode control si backstepping control**. Mai jos se prezinta diagramele de baza pentru implementarea acestor modalitati de conducere care permit urmărirea traiectoriei impuse, evitarea obstacolelor si reducerea miscarilor nedorite ale platformei.



Pentru a avea feedback al spatiului din imediata apropiere se va utiliza senzorul ultrasonic cu functie de autoinvatare, tip U-GAGE® T30UX cu compensare a temperaturii.

4. Sursa on-board de energie electrica permite deplasarea autonoma a platformei. Această sursă va alimenta toate sistemele electrice și electronice.

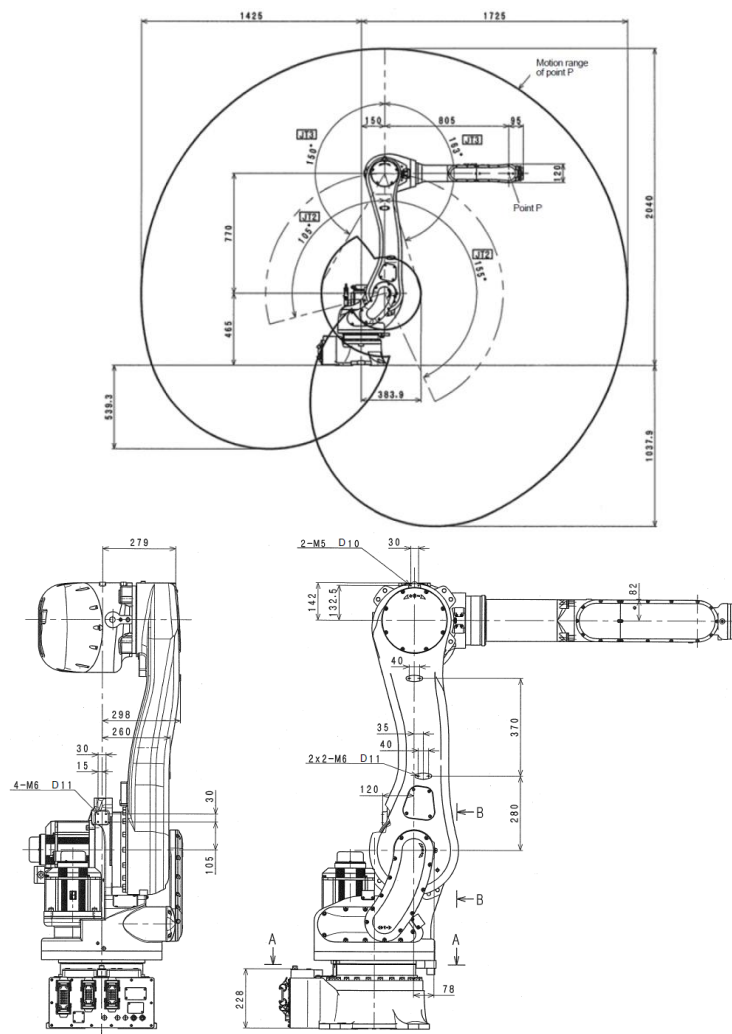
Bratul robotic

Modelul de brat robotic este tipul RS020N-A produs de Kawasaki, Japonia.

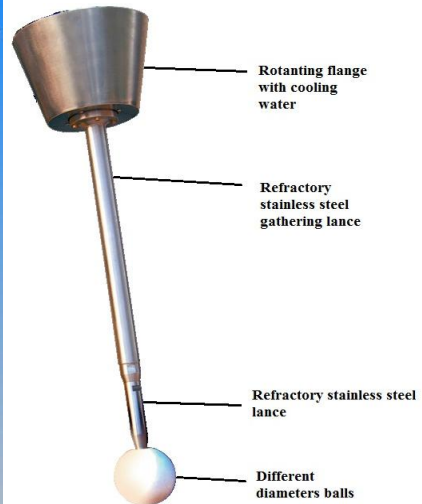
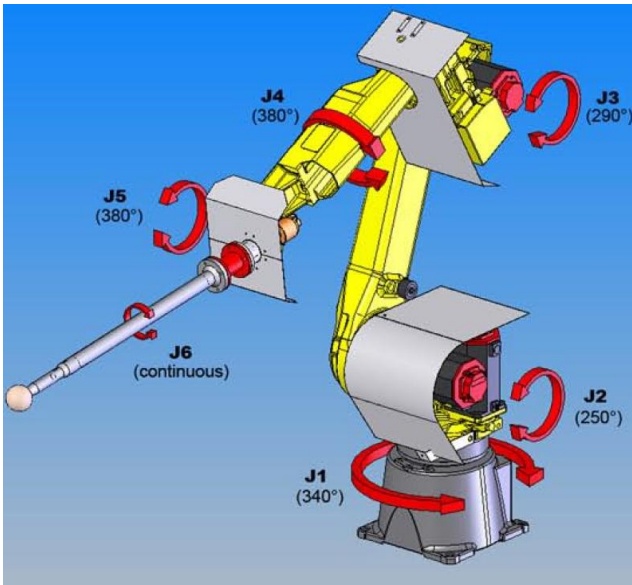
Acest model de robot este articulat, cu 6 axe de mișcare liberă (6-DOF). Tabelul de mai jos prezintă posibilitățile de mișcare și viteze de deplasare pe fiecare axă.

Sarcina maximă pe care o poate manevra la capătul axei J6 este de 20kg incluzând dispozitivul de colectare a sticlei topite din cuptor. Permite o arie de lucru în jurul bazei de 10 mp.

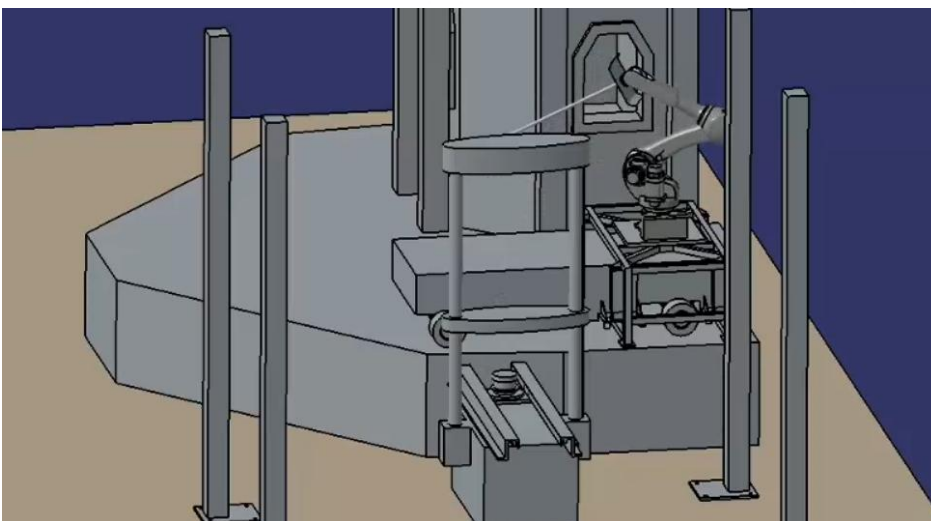
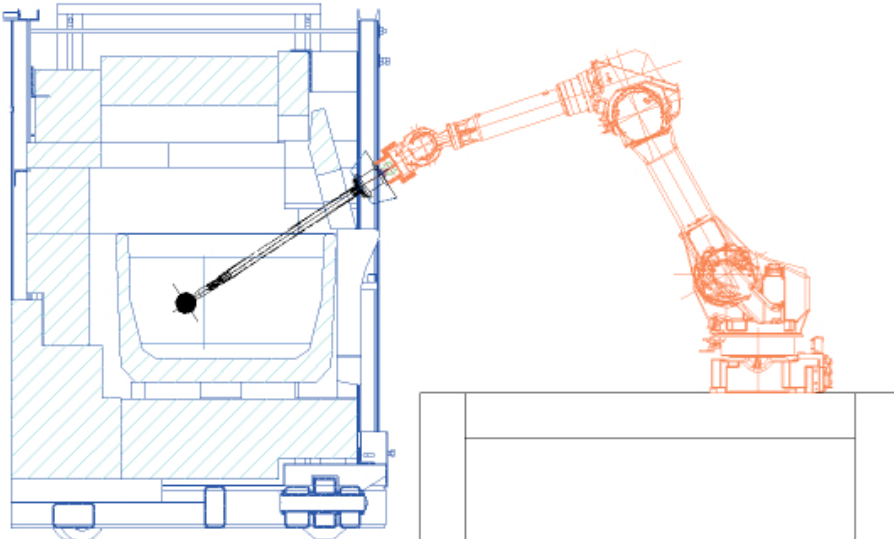
Type	Articulated Robot		
Degree of Freedom	6		
Motion Range and Maximum Speed	JT	Motion Range	Max. Speed
	1	$\pm 180^\circ$	190 °/s
	2	+155° to -105°	205 °/s
	3	+150° to -163°	210 °/s
	4	$\pm 270^\circ$	400 °/s
	5	$\pm 145^\circ$	360 °/s
6	$\pm 360^\circ$	610 °/s	
Max. Payload	20 kg		
Wrist Load Capacity	JT	Torque	Moment of Inertia
	4	45.0 N·m	0.9 kg·m ²
	5	45.0 N·m	0.9 kg·m ²
6	29.0 N·m	0.3 kg·m ²	
Repeatability	± 0.05 mm		
Mass	230 kg		
Acoustic noise	< 70 dB (A)*		

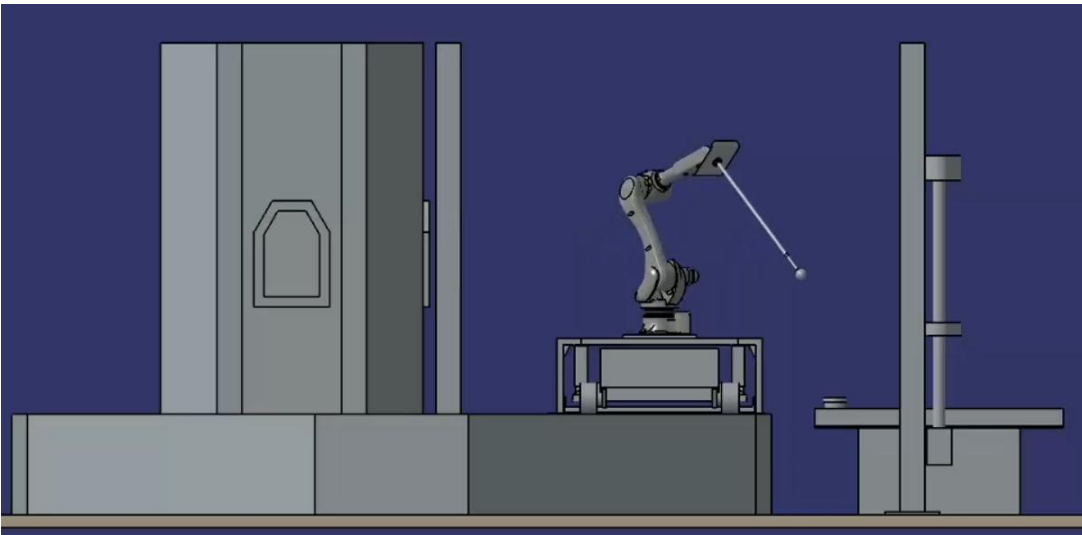


Pentru îndeplinirea rolului tehnologic de extragere a sticlei topite dintr-un cuptor de menținere se va utiliza o lance echipată cu bilă ceramică. Sticla aderă pe suprafața tablei care se rotește odată cu lancea putând să preia cantități de până la 500 g (în cazul acestui proiect) necesare pentru alimentarea unei preese de manufacturare obiecte din sticlă (farfurii, boluri) pentru producția de serie mare.



S-au efectuat simulări ale condițiilor reale din fabrica de sticlă pentru a se stabili cinematica axelor brațului robotului (prezentat mai jos).





Conducerea bratului robotic se face cu ajutorul controlerului dedicat tip E91, Kawasaki, capabil să coordoneze 6 axe, cu servosisteme digitale, alimentat la 230 Vca, 50Hz. În echiparea acestui controller este și consola Teaching Pad pentru implementarea cinematicii în funcție de aplicație.



Activitatea II.5 Trimiterea spre acceptare și publicare de articole la următoarele conferințe: CIS&RAM 2015 și ICSTCC 2015

Razvan Solea, Adrian Filipescu, Adriana Filipescu Jr. Eugenia Minca, Silviu Filipescu, Wheelchair Control and Navigation Based on Kinematic Model and Iris Movement, Proceedings of the 2015 7th IEEE International Conference on Robotics, Automation and Mechatronics (CIS&RAM), 15 – 17 July 2015 Angkor Wat, Cambodia, IEEE Catalog Number: CFP15835-CDR, ISBN: 978-1-4673-7336-4, pp:78-83.

Minca E., Filipescu, A., Dragomir O., Coanda, H., G., Dragomir F., Cycle Time Optimization of a Reversible A/DML Served by a Mobile Robotic System, Proceedings of the 19th IEEE, International Conference on System Theory, Control and Computing, ICSTCC 2015 14-16, Oct. Cheile Gradistei, Romania, 2015, pp.99-104, ISBN: 978-1-4799-8481-7 ©2015 IEEE.

Filipescu, A., Minca E., Voda A., Dumitrascu B., Filipescu A., Jr., Ciubucciu G., Sliding-Mode Control and Sonar Based Bubble Rebound Obstacle Avoidance for a WMR, Proceedings of the 19th IEEE, International Conference on System Theory, Control and

Computing, ICSTCC 2015 14-16, Oct. Cheile Gradistei, Romania, 2015, pp.105-110,
ISBN: 978-1-4799-8481-7©2015 IEEE.