

MINISTERUL EDUCAȚIEI



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

# SMART WAREHOUSE INVENTORY MANAGEMENT

PROIECT DE SEMESTRUL 1, ANUL 3

Student: **Dan Ovidiu**

**Pașca Adrian Alexandru**

Disciplina: **Sisteme bazate pe Cunoaștere**

**2025**

# Cuprins

<b>1</b>	<b>INTRODUCERE (1-5 PAG)</b> .....	<b>2</b>
1.1	CONTEXT GENERAL .....	2
1.2	OBIECTIV .....	2
1.3	SPECIFICAȚII .....	2
<b>2</b>	<b>CUNOAȘTEREA ȘI ANALIZA SETULUI DE DATE (1-5 PAG)</b> .....	<b>3</b>
<b>3</b>	<b>PRE-PROCESAREA SETULUI DE DATE (1-5 PAG)</b> .....	<b>5</b>
<b>4</b>	<b>MODELAREA SISTEMULUI (1-10 PAG)</b> .....	<b>7</b>
<b>5</b>	<b>CONCLUZII (1-3 PAG)</b> .....	<b>12</b>
5.1	REZULTATE OBȚINUTE.....	12
5.2	DIRECȚII DE DEZVOLTARE.....	13
<b>6</b>	<b>BIBLIOGRAFIE (1-2 PAG)</b> .....	<b>14</b>
<b>7</b>	<b>REGULI DE FORMATARE</b> .....	<b>15</b>
7.1	FORMATAREA PAGINII .....	15
7.2	TITLURI ȘI STILURI .....	15
7.3	FIGURI, TABELE ȘI ECUAȚII .....	16
7.3.1	<i>Figuri</i> .....	16
7.4	TABELE .....	16
7.5	ECUAȚII .....	16
7.6	REFERINȚE BIBLIOGRAFICE .....	17

# 1 Introducere

## 1.1 Context general

Automatizarea proceselor de gestionare a stocurilor este esențială pentru eficientizarea fluxurilor de lucru în depozitele moderne. Proiectul "Smart Warehouse Inventory Management" are ca scop dezvoltarea a 2 roboți autonomi capabili să realizeze sarcini specifice, cum ar fi deplasarea controlată și detectarea traseelor, contribuind astfel la optimizarea operațiunilor de gestionare a depozitelor.

## 1.2 Obiective

- Colectarea datelor de performanță a motoarelor pentru analiza comportamentului.
- Pre-procesarea datelor pentru eliminarea zgomotului.
- Identificarea funcțiilor de transfer pentru ambele motoare.
- Proiectarea și implementarea unui regulator PI folosind metoda Ziegler-Nichols.
- Implementarea unui algoritm de lane detection pentru navigația autonomă a robotului

## 1.3 Specificații

- Controlul a două motoare DC utilizând 2 regulatoare PI.
- Utilizarea senzorilor optici pentru detectarea traseului marcat.
- Medii de dezvoltare: Python (pentru pre-procesare), MATLAB (pentru modelare și control), Arduino (testarea funcționalităților).

## 2 Cunoașterea și analiza setului de date

Primul pas a fost să aplicăm diferiți factori de umplere pe fiecare motor al MainBot-ului. Cele trei valori sunt 85, 128 și 255, care reprezintă tensiunea aplicată pe fiecare DC motor în parte, care sunt alimentate de 4 baterii de 3.7V, dispuse câte două în serie și apoi conectate în paralel, care oferă o tensiune totală de 7.4V.

1. Pentru PWM = 85:

$$V_{out} = \frac{85}{255} \times 7.4 \approx 2.47 \text{ V}$$

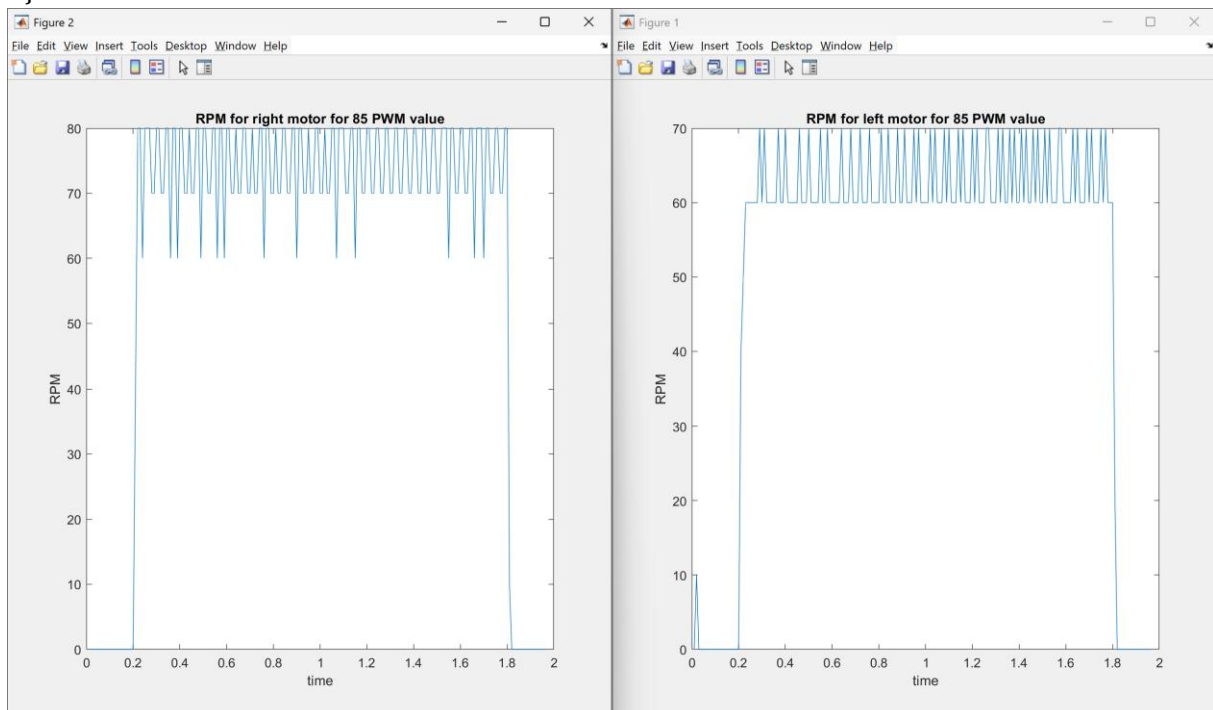
2. Pentru PWM = 128:

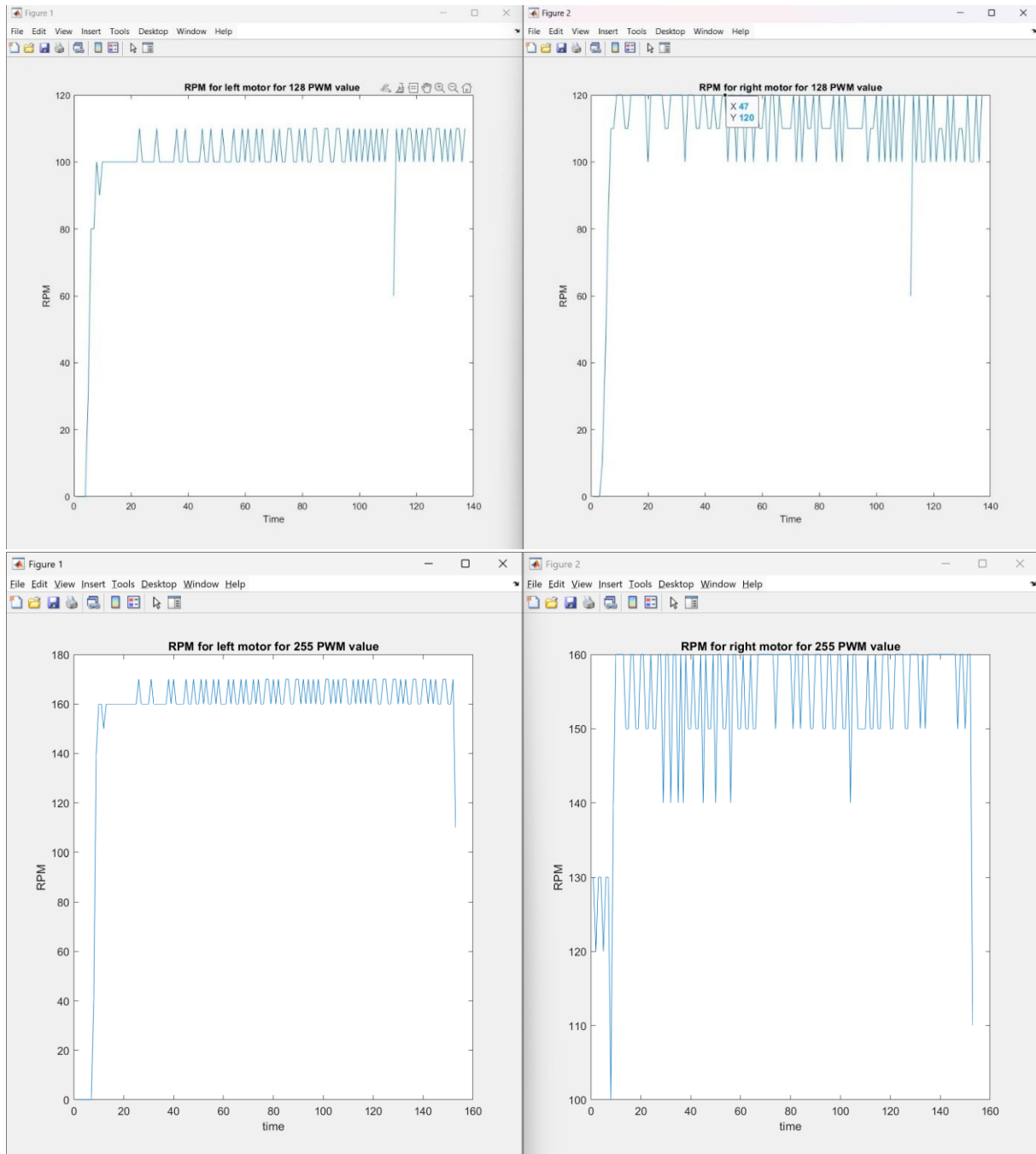
$$V_{out} = \frac{128}{255} \times 7.4 \approx 3.71 \text{ V}$$

3. Pentru PWM = 255:

$$V_{out} = \frac{255}{255} \times 7.4 = 7.4 \text{ V}$$

Fiecare set de date a fost salvat într-un fișier excel separat, care ulterior a fost importat în Matlab pentru vizualizare. Datele au fost preluate cu o perioadă de eșantionare de 0.01s.





Am început prin a înțelege setul de date. Având doar trei caracteristici, nu a fost necesar să reducem dimensionalitatea setului de date.

	Speed Left	Speed Right	Time
count	196.000000	196.000000	196.000000
mean	51.326531	60.765306	0.975000
std	24.479725	29.455329	0.567245
min	0.000000	0.000000	0.000000
25%	60.000000	60.000000	0.487500
50%	60.000000	70.000000	0.975000
75%	60.000000	80.000000	1.462500
max	70.000000	80.000000	1.950000

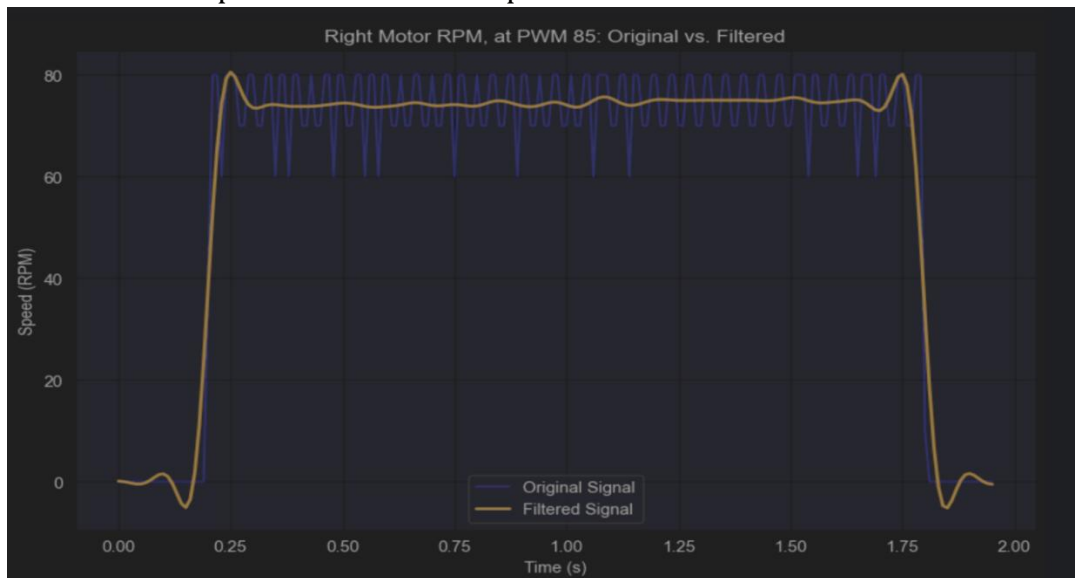
### 3 Pre-procesarea setului de date

Pre-procesarea datelor a fost realizată în PhyCharm/Jupyter Notebook.

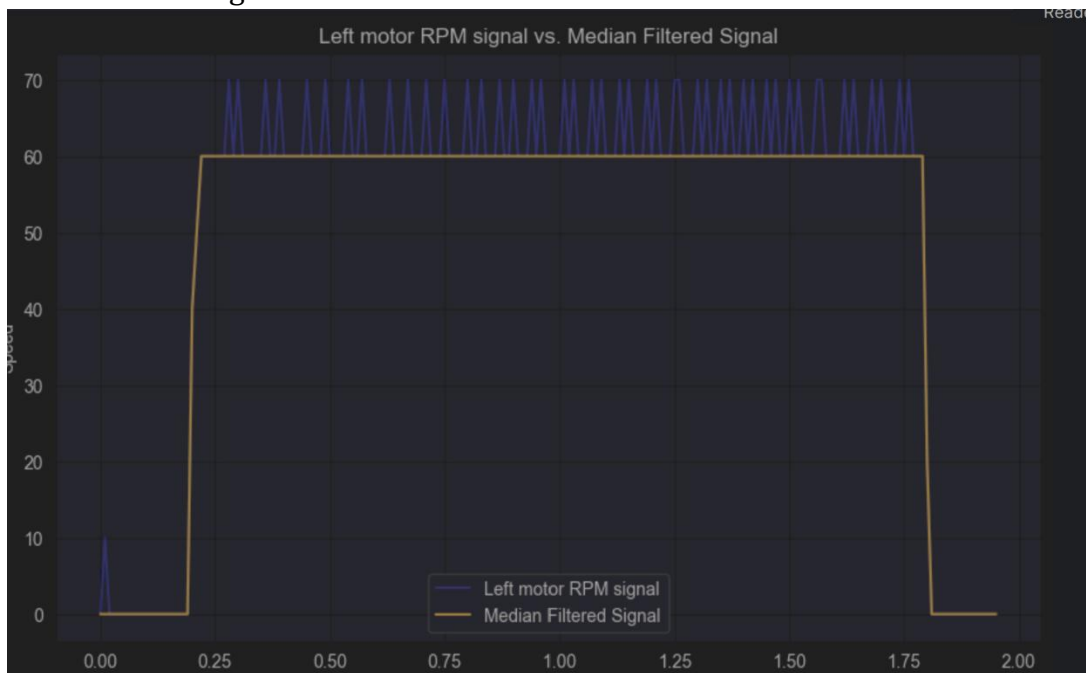
Următorul pas a fost aplicarea tehnicilor de eliminare a zgomotelor. Pentru fiecare PWM ales și pentru fiecare motorăș am folosit tehnica care s-a potrivit cel mai bine. Am separat datele în 2 categorii: date pentru identificare și date pentru antrenare/validare. Această caracteristică ne-a ajutat ulterior pentru realizarea modelului matematic.

#### Identificare

- Motorul DC drept la PWM = 85 – lowpass filter

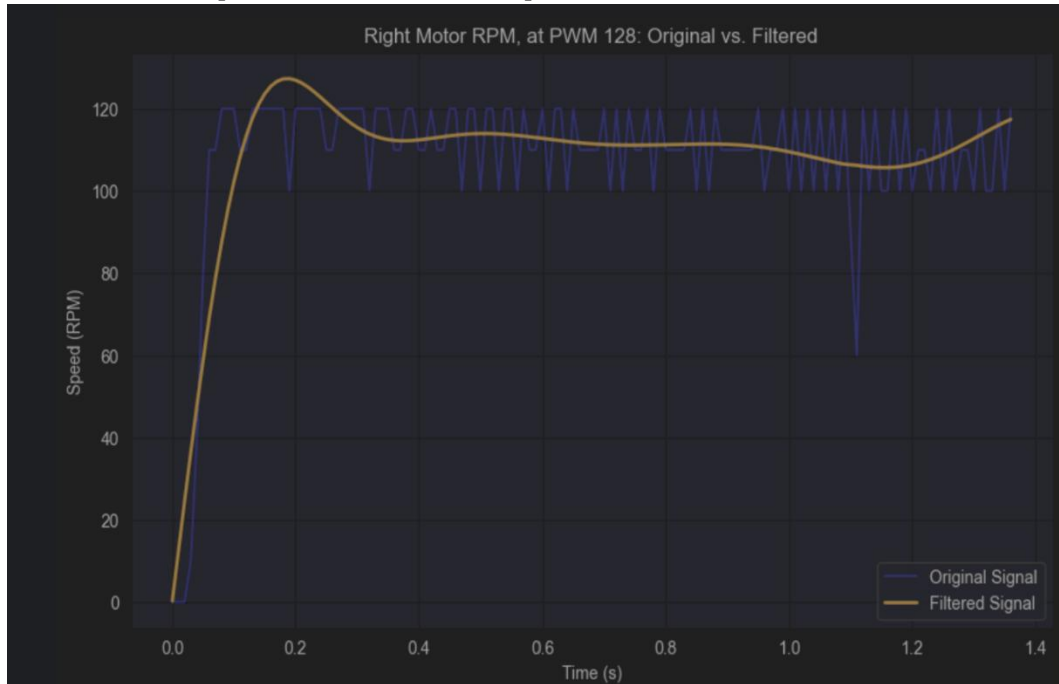


- Motorul DC stâng la PWM = 85 – median filter

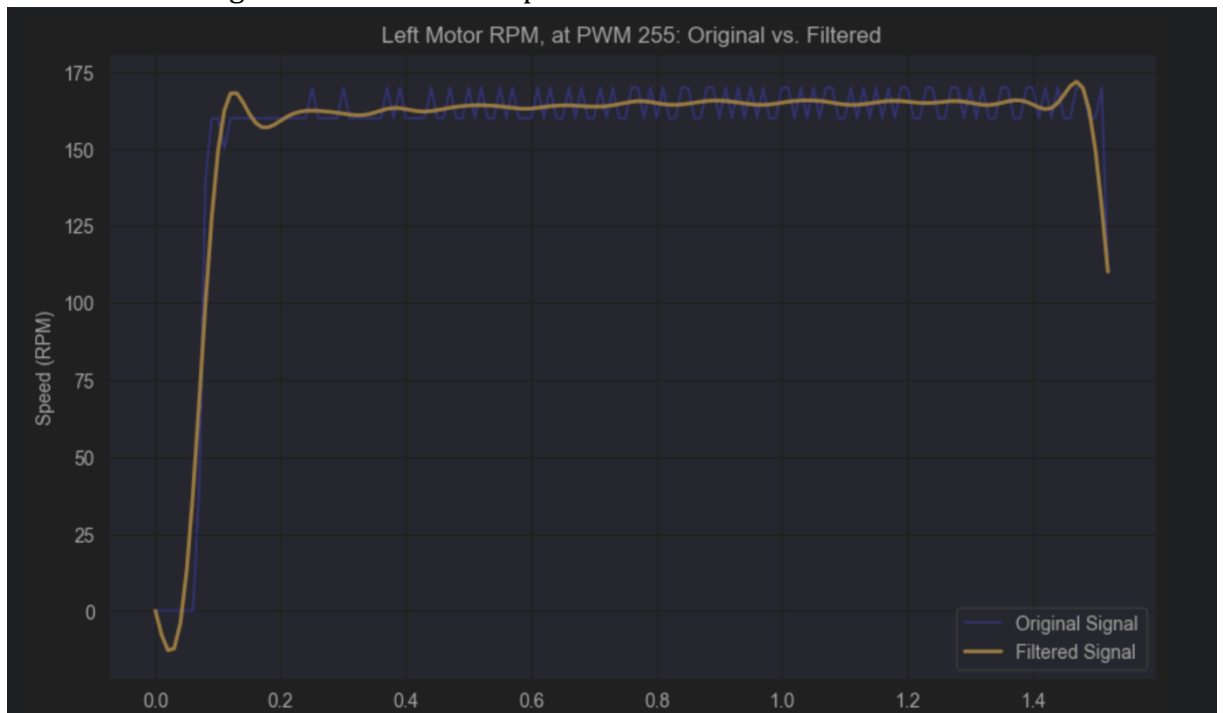


## Validare

- Motorul DC drept la PWM = 128 – lowpass filter



- Motorul DC stâng la PWM = 255 – lowpass filter



După filtrare, datele au fost importate înapoi în Matlab pentru identificare și validarea modelului matematic.

## 4 Modelarea sistemului

Pentru modelarea sistemului ne-am propus să aplicăm răspunsul la treaptă, din care va rezulta un model parametric, de tip cutie gri (parțial cunoscut în avans). Funcția de transfer are forma:

$$H(s) = \frac{K}{T \cdot s + 1}$$

### 4.1 Identificarea funcțiilor de transfer

Am aplicat următorul algoritm pentru identificarea parametrilor:

#### Algoritm general

- 1 Citește  $u_0$ ,  $y_0$ ,  $u_{ss}$ ,  $y_{ss}$ , valorile inițiale și în regim staționar ale intrării și ieșirii. Calculează  $K = \frac{y_{ss} - y_0}{u_{ss} - u_0}$ .
- 2 Citește timpul  $t_0$  unde are loc treapta, și  $t_1$  unde ieșirea urcă la 0.632 din diferență. Calculează  $T = t_1 - t_0$ .

În Matlab, am utilizat următoarea secvență de cod:

- pentru motorul stâng:

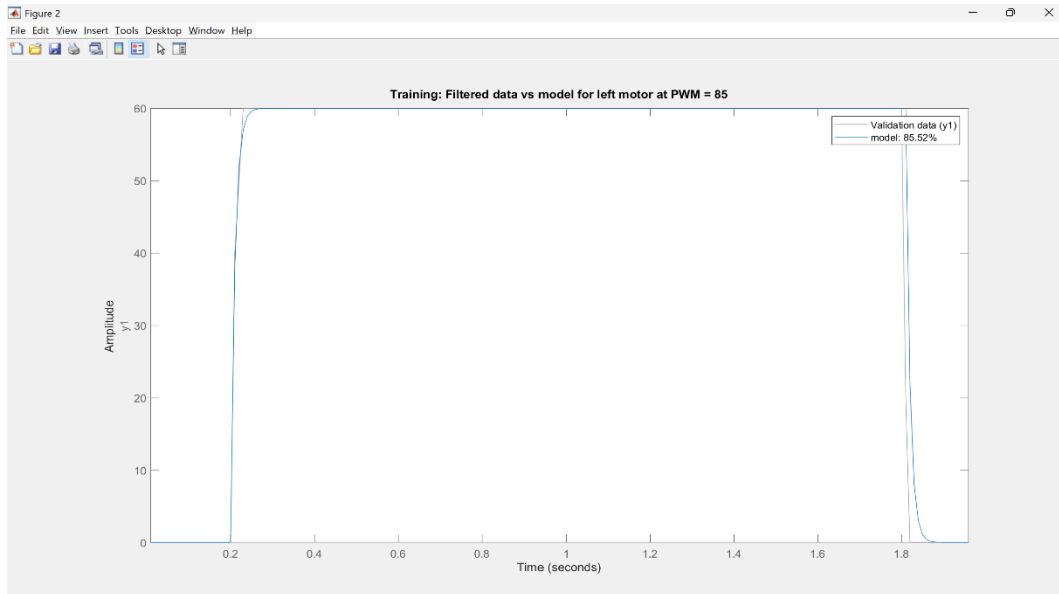
```

30 %% left motor model - identificare on PWM 85 (median filtered)
31 u = zeros(1,196);
32 u(20:180) = 2.47;
33
34 u_id = u;
35 y_id = filtered_signal_left;
36
37 yss = 60;
38 uss = 2.47;
39
40 K = yss / uss;
41 t0 = 20;
42 t1 = 21;
43 T = (t1 - t0) / 100;
44
45 H = tf(K,[T,1]);
46
47 y_pred = lsim(H,u_id,time);
48
49 data_id = iddata(filtered_signal_left(:),u_id(:),0.01);
50 model = iddata(y_pred(:),u_id(:),0.01);
51 figure, compare(data_id,model), title("Training: Filtered identification data vs model for left motor at PWM = 85")
52 % K = 24.2915, T = 0.01

```



S-a obținut modelul cu parametrii  $K = 24.2915$  și  $T = 0.01$ . Comparând datele cu modelul obținem:



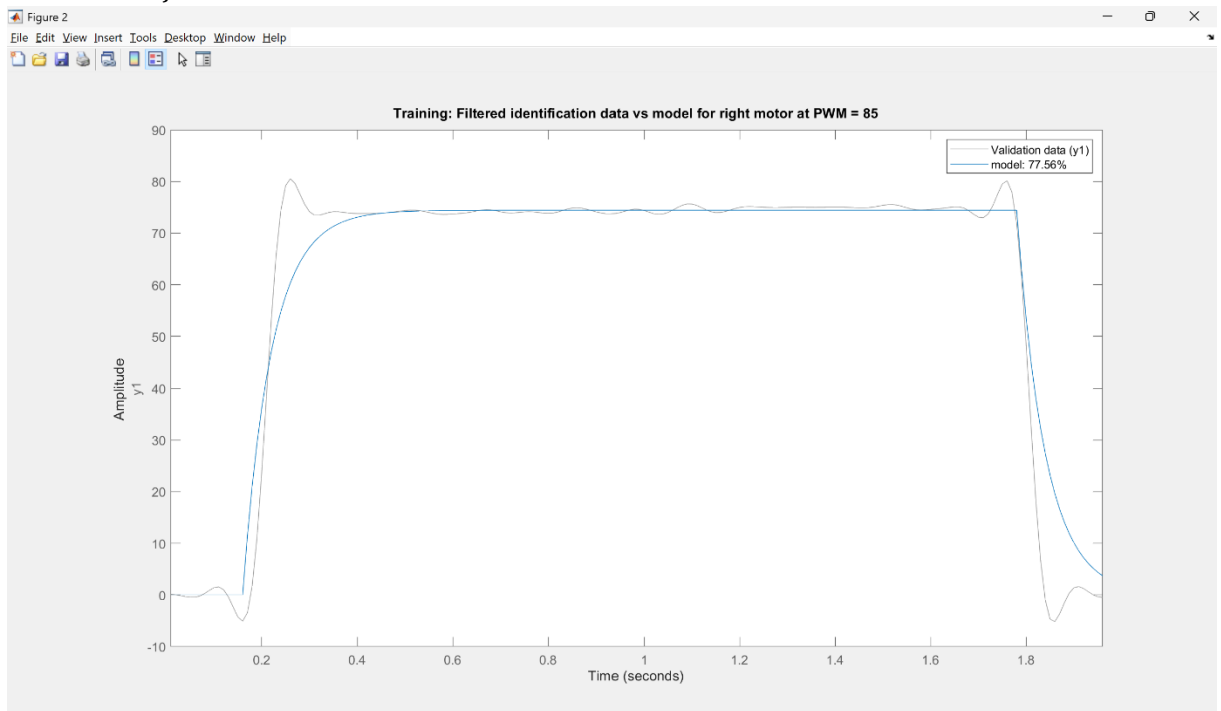
- pentru motorul drept:

```

10 %% identificare right motor PWM = 85
11 close all
12 u = zeros(1,196);
13 u(16:177) = 2.47;
14
15 u_id = u;
16 y_id = filtered_signal_right;
17 data_id = iddata(y_id(:),u_id(:),0.01);
18 model_arx = arx(data_id,[1 1 0]);
19
20 yss = mean(y_id(30:173));
21 uss = 2.47;
22
23 K = yss / uss;
24 t0 = 16;
25 t1 = 22;
26 T = (t1 - t0) / 100;
27
28 H = tf(K,[T,1]); % K = 30.1339 & T = 0.06
29
30 y_pred = lsim(H,u_id,time);
31
32 data_id = iddata(filtered_signal_right(:),u_id(:),0.01);
33 model = iddata(y_pred(:),u_id(:),0.01);
34 figure, compare(data_id,model), title("Training: Filtered identification data vs model for right motor at PWM = 85")

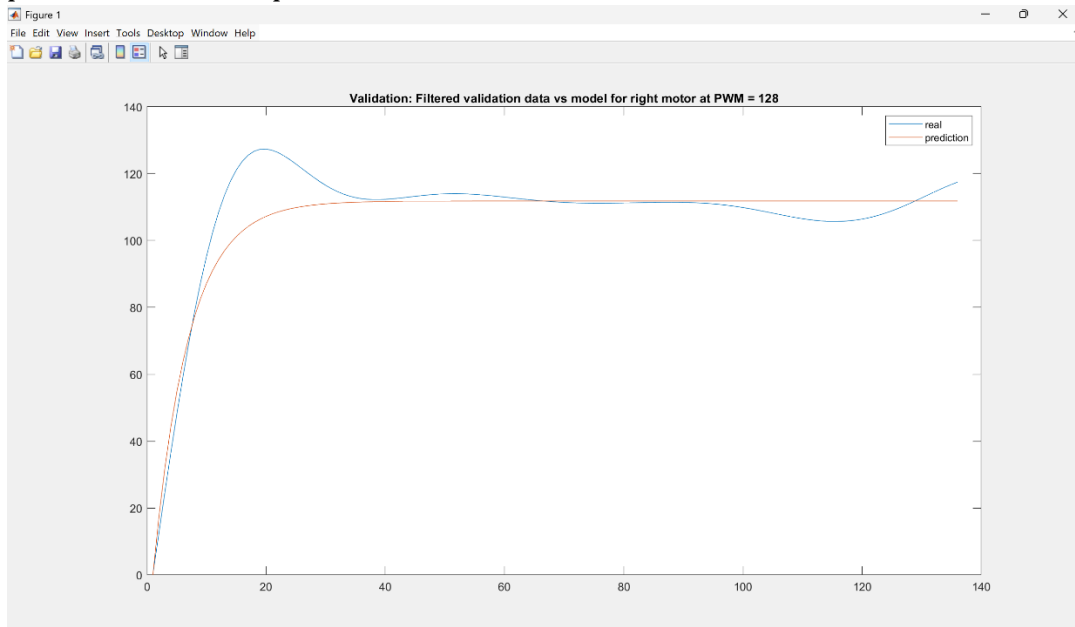
```

S-a obținut modelul cu parametrii  $K = 30.1339$  și  $T = 0.06$ . Comparând datele cu modelul obținem:

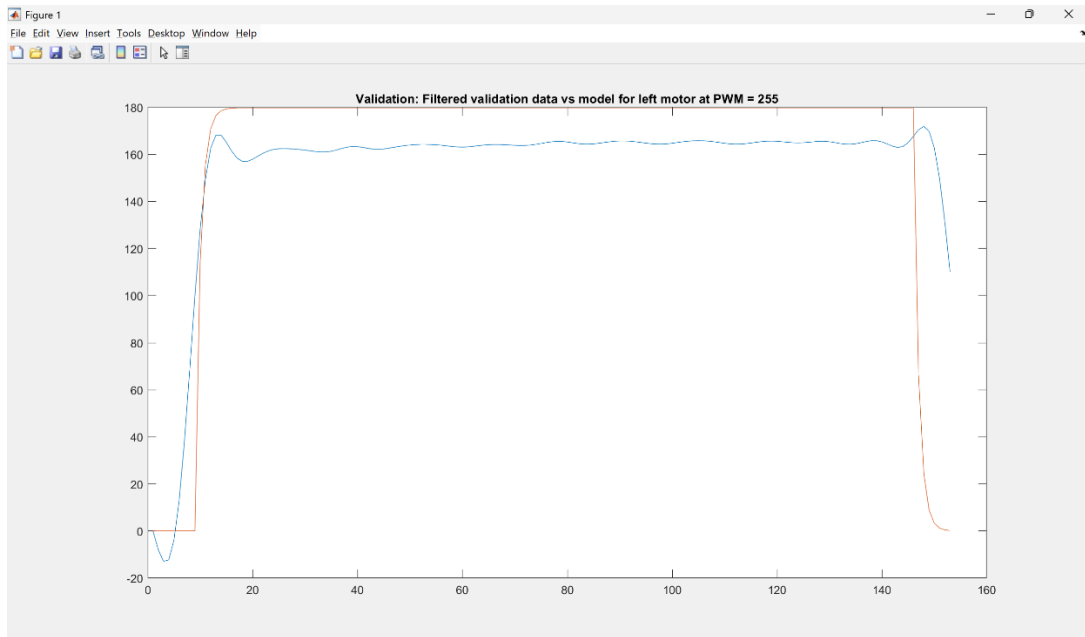


Mai apoi, am validat modelele pe celelalte două grafice:

- pentru motorul drept



- pentru motorul stâng



## 4.2 Proiectarea regulatorului

Am proiectat un regulator de tip PI. Am observat că timpul de răspuns este unul foarte bun pentru ambele modele, deci efectul derivativ poate fi eliminat.

Metoda folosită pentru determinarea reguletoarelor a fost Ziegler-Nichols, care este o tehnică rapidă și eficientă de proiectare. Am dus la limita de stabilitate sistemul, unde am obținut un  $K_{cr}$  și un  $T_{cr}$ , iar mai departe am folosit tabelul:

Type	$k_p$	$T_i$	$T_d$
P	$0.5k_c$		
PI	$0.4k_c$	$0.8T_c$	
PID	$0.6k_c$	$0.5T_c$	$0.125T_c$

(b) Frequency response method

Parametrii reguletoarelor de tip PI sunt:

- pentru motorul drept:

$$H_c = \frac{1}{K_p + K_i * \frac{1}{s}}$$

with  $K_p = 0.0646$ ,  $K_i = 2.93$

- pentru motorul stâng:

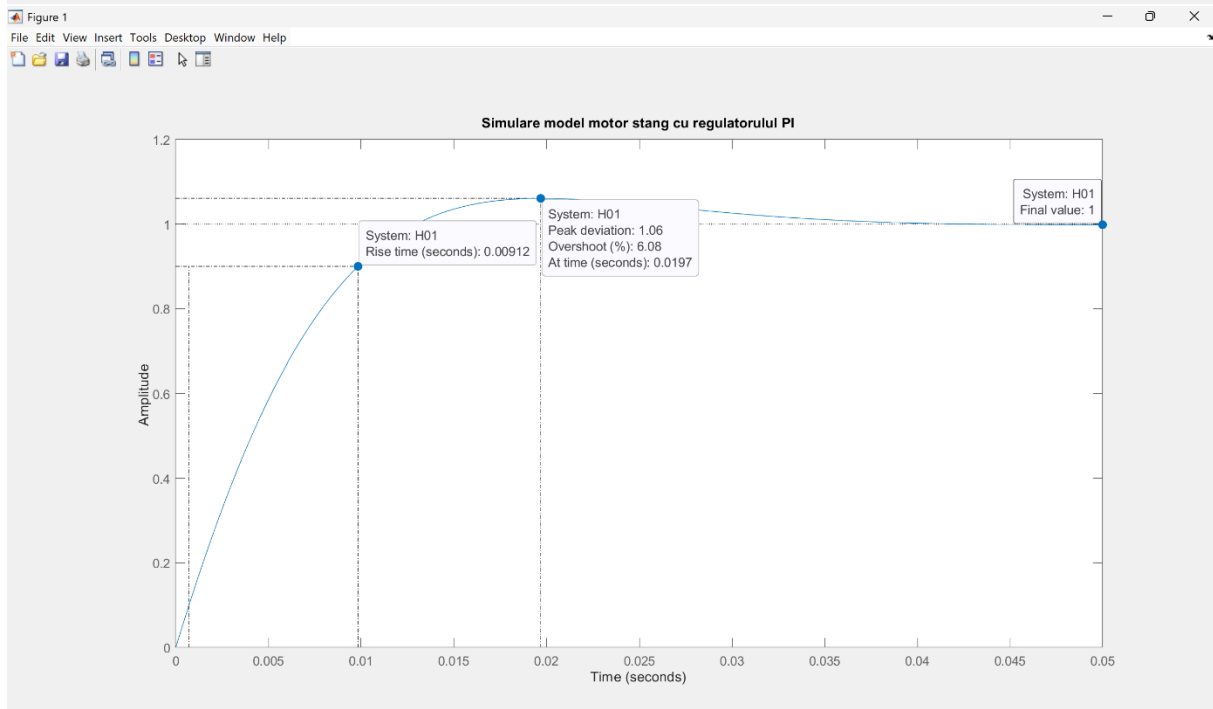
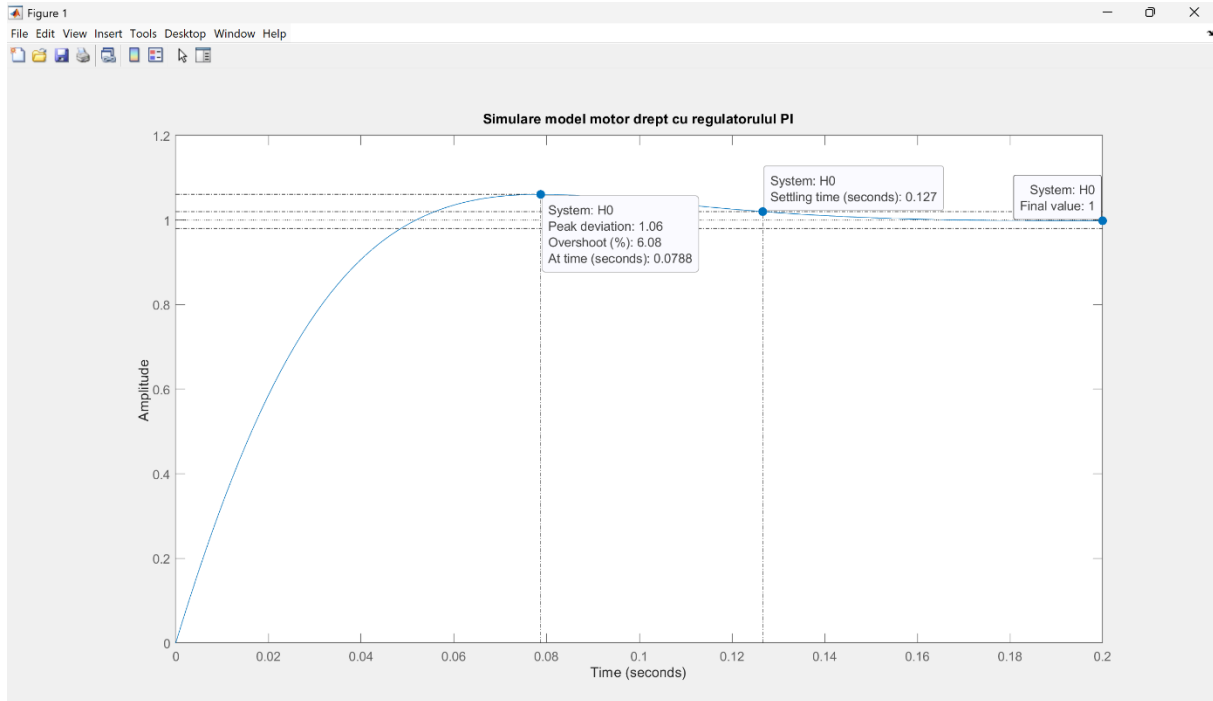
$$C = \frac{1}{K_p + K_i * \frac{1}{s}}$$

with  $K_p = 0.0589$ ,  $K_i = 10.7$

# 5 Concluzii

## 5.1 Rezultate obținute

Am evaluat răspunsul fiecărui motor DC controlat de PI la semnalul de tip treaptă și am obținut următoarele rezultate:



### 5.1.1 Alte funcționalități

Am utilizat regulatoarele PI pentru a dezvolta un sistem de lane detection bazat pe doi senzori digitali optici. Acesta funcționează pe un traseu închis la culoare, marcat cu linii albe.

- **Ambii senzori detectează linii:** robotul se deplasează înainte cu viteză constantă.
- **Doar senzorul drept detectează linia:** motorul stâng se oprește, iar cel drept rămâne activ, robotul virând spre stânga.
- **Doar senzorul stâng detectează linia:** robotul virează spre dreapta, prin oprirea motorului drept și menținerea activă a celui stâng.

### 5.2 Direcții de dezvoltare

- Dezvoltarea regulatoarelor pentru lane detection pentru a păstra o traiectorie optimă
- Adăugarea de noi senzori pentru o navigație mai precisă
- Implementarea brațului robotic pentru a muta obiecte
- Conectarea MainBot-ului cu CarryBot-ul

## 6 Bibliografie

<https://www.mathworks.com/help/control/ug/proportional-integral-derivative-pid-controllers.html>

[https://en.wikipedia.org/wiki/Ziegler%E2%80%93Nichols\\_method](https://en.wikipedia.org/wiki/Ziegler%E2%80%93Nichols_method)

<https://www.mathworks.com/help/control/ref/dynamicsystem.step.html>

<https://www.geeksforgeeks.org/exploratory-data-analysis-in-python/>

[https://busoniu.net/teaching/sysid2024/sysid24ro\\_transient\\_handout.pdf](https://busoniu.net/teaching/sysid2024/sysid24ro_transient_handout.pdf)

## 7 Reguli de formatare

### 7.1 Formatarea paginii

- Dimensiunea paginii: A4
- Margini: 2.5 cm (sus, jos, stânga, dreapta)
- Antet și subsol: 1.27 cm de la marginea paginii
- În antetul paginii (header): titlul capitolului, centrat, stil: Header\_style
- În subsolul paginii: numărul paginii, centrat

### 7.2 Titluri și stiluri

Titlurile capitolelor și subcapitolelor se marchează cu stilurile Heading 1 – 4, conform documentului model anexat în format Word. Descrierea stilurilor utilizate în document este prezentată în Tabelul 7.1.

*Tabelul 7.1. Stiluri utilizate în acest document*

Nr.	Stil	Utilizat pentru	Format
1	Normal	Text normal	Font: (Default) Cambria, 12 pt, Justified, Line spacing: Multiple 1.1 li, Space After: 6 pt
2	Titlu	Titlul proiectului, prima pagină	Font: 24 pt, Small caps, Centered Line spacing: single, Space Before: 126pt, After: 0 pt,
3	Titlu2	Titlul proiectului, pagina de prezentare	Font:14pt, Bold, Centered
4	Heading 1	Titlurile capitolelor (nivel 1)	Font: 24 pt, Indent: Left: 0 cm Hanging: 0.76 cm, Space Before: 24pt, After: 12pt
5	Heading 2	Titlurile subcapitolelor (nivel 2)	Font: 14 pt, Bold, Indent: Left: 0 cm Hanging: 1.02 cm, Space Before: 18pt, After: 12pt
6	Heading 3	Titlurile secțiunilor (nivel 3)	Font: Bold, Indent: Left: 0 cm Hanging: 1.27 cm, Space Before: 6 pt, After: 6pt
7	Heading 4	Titlurile secțiunilor (nivel 4)	Font: Italic, Indent: Left: 0 cm Hanging: 1.52 cm, Space Before: 2 pt, After: 0 pt
8	Caption	Legenda figurilor și tabelelor	Font: Italic, Font color: Text 1, Line spacing: single, Space After: 10 pt,



Nr.	Stil	Utilizat pentru	Format
9	Header_style	Antetul paginii	Font: 10 pt, Italic, Centered, Border: Bottom: (Single solid line, Background 1, 0.5 pt Line width)

## 7.3 Figuri, tabele și ecuații

### 7.3.1 Figuri

Figurile se inserează în text centrate, cu etichetă de numerotare și legendă (Caption) în partea de jos a figurii. Numărul figurii include și numărul capitolului, după exemplul prezentat în Figura 7.1.

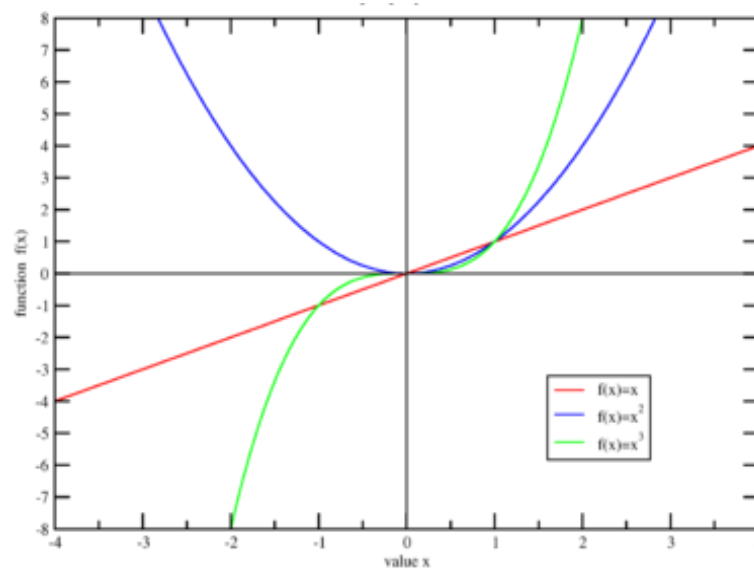


Figura 7.1. Figură exemplu, stil: Caption

## 7.4 Tabele

Tabelele se inserează în text centrate, cu etichetă și legendă (Caption) în partea de sus a tabelului, aliniată la stânga. Numărul tabelului include și numărul capitolului, după cum este prezentat, de exemplu, în Tabelul 7.1.

## 7.5 Ecuații

Ecuațiile se inserează în text centrate, cu numerotare în partea dreaptă. Numărul ecuației include și numărul capitolului, conform exemplului din relația (5.1).

$$(x + a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k} \quad (5.1)$$

## 7.6 Referințe bibliografice

Se recomandă ca citarea referințelor bibliografice să fie făcută în formatul IEEE.

În secțiunea Bibliografie sunt prezentate exemple pentru: o citare a unui capitol dintr-o carte [1], un articol publicat într-o revistă [2] și un articol publicat la o conferință [3].

Detalii cu privire la formatul citării diverselor tipuri de referințe pot fi găsite în [4] sau [5].

Referințele bibliografice se pot insera în text utilizând facilitățile Word de a adăuga surse și bibliografie unui document (References -> Citations & Bibliography). Dacă formatul IEEE pentru bibliografie nu este instalat implicit în Word, se poate descărca gratuit de la:

<https://bibword.codeplex.com/wikipage?title=Styles&referringTitle=Home>

Instrucțiunile de instalare pentru diferite versiuni de Word se pot obține de la aceeași adresă.