

Ψηφιακά Συστήματα Μετρήσεων

Arduino # Raspberry Pi
Processing # Python # MATLAB

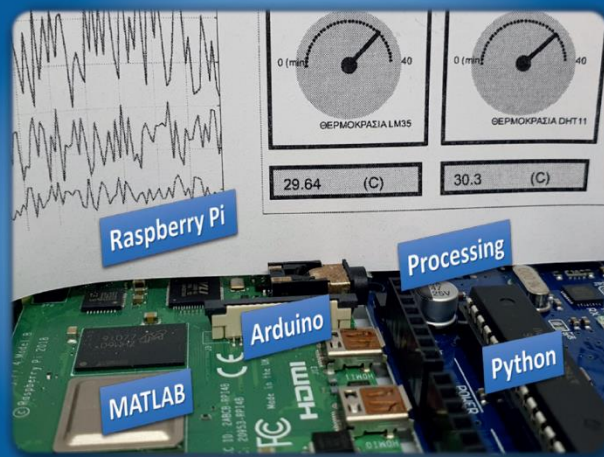
Το βιβλίο αυτό «απαντά» στις πραγματικές ανάγκες ενός μηχανικού, στην κατεύθυνση της ανάπτυξης ενός λειτουργικού και ολοκληρωμένου ψηφιακού συστήματος μετρήσεων. Παρουσιάζει τις επικρατέστερες τεχνολογίες και μεθοδολογίες για την ανάπτυξη των εφαρμογών. Δηλαδή, συγκεντρώνει γνώσεις με μια ενιαία αντίληψη και όχι αποσπασματικά, όπως έχουν συνηθίσει οι μηχανικοί σήμερα, που καλούνται να ανατρέξουν σε τελείως διαφορετικά βιβλία μεταξύ τους, μη γνωρίζοντας ακριβώς ποια εργαλεία θα πρέπει να χρησιμοποιήσουν.

Το βιβλίο είναι εστιασμένο στην ανάπτυξη λογισμικού, εκεί δηλαδή που βρίσκεται ο πυρήνας των σύγχρονων ψηφιακών συστημάτων μέτρησης.

Το βιβλίο συνοδεύεται από πλούσιο υλικό που είναι διαθέσιμο στην ιστοσελίδα panosparazoglou.gr

ΠΑΝΑΓΙΩΤΗΣ ΠΑΠΑΖΟΓΛΟΥ

Ψηφιακά Συστήματα Μετρήσεων



Διαφάνειες

Υλικό βιβλίου

Περισσότερο υλικό στο
panosparazoglou.gr

Κεφάλαιο 9



Απεικόνιση και επεξεργασία μετρήσεων σε υπολογιστή με χρήση MATLAB



Μερικά χαρακτηριστικά MATLAB

- Μαθηματικούς υπολογισμούς (απλά και προχωρημένα μαθηματικά, στατιστική, κλπ)
- Γραφική απεικόνιση
- Ανάπτυξη κώδικα
- Ειδικά πακέτα (Toolbox) για τα σημαντικότερα επιστημονικά πεδία των μηχανικών (π.χ. βάσεις δεδομένων, επεξεργασία εικόνας, αεροναυπηγική, τηλεπικοινωνίες, εφαρμογές HDL, αυτόματος έλεγχος, βιοπληροφορική, κλπ)
- Προσομοίωση και μελέτη συστημάτων σε επίπεδο μπλοκ διαγράμματος (Simulink), κλπ

Εναλλακτικό δωρεάν λογισμικό
Octave

Η βασική δομή του πίνακα

Δημιουργία πίνακα

`a=[10 20 30 40 50 60 70 80 90 100]`

```
clear;  
i=1;  
for x=10:10:100  
    a(i)=x;  
    i=i+1;  
end
```

**Συμβατικός
Προγραμματισμός
στο MATLAB**

```
a=[10:10:100]
```

**Προγραμματισμός
MATLAB**

Παραδείγματα (1)

```
>> b=[10:10:100]
```

```
b =
```

```
10 20 30 40 50 60 70 80 90 100
```

```
>> b=[10:5.5:100]
```

```
b =
```

```
Columns 1 through 6
```

```
10.0000 15.5000 21.0000 26.5000 32.0000 37.5000
```

```
Columns 7 through 12
```

```
43.0000 48.5000 54.0000 59.5000 65.0000 70.5000
```

```
Columns 13 through 17
```

```
76.0000 81.5000 87.0000 92.5000 98.0000
```

```
>> b'
```

```
ans =
```

```
10
```

```
20
```

```
30
```

```
40
```

```
50
```

```
60
```

```
70
```

```
80
```

```
90
```

```
100
```



Παραδείγματα (2)

```
>> a=[4 5.321 43 12 4.99 5]
```

```
a =
```

```
4.0000 5.3210 43.0000 12.0000 4.9900 5.0000
```

```
>> a=[4 5.321 ; 43 12 ; 4.99 5]
```

```
a =
```

```
4.0000 5.3210  
43.0000 12.0000  
4.9900 5.0000
```



Παραδείγματα (3)

Δημιουργία πίνακα:

- α) πρώτη γραμμή από το 0 έως το 10 με βήμα 2
- β) δεύτερη γραμμή από το 10 έως το 0 με βήμα 2
- γ) τρίτη γραμμή από το 0.3 έως το 0.8 με βήμα 0.1

```
>> a=[0:2:10 ; 10:-2:0 ; 0.3:0.1:0.8]
```

```
a =
```

0	2.0000	4.0000	6.0000	8.0000	10.0000
10.0000	8.0000	6.0000	4.0000	2.0000	0
0.3000	0.4000	0.5000	0.6000	0.7000	0.8000

Παραδείγματα (4)

```
>> a=zeros(5,1)
```

```
a =
```

```
0  
0  
0  
0  
0
```

```
>> a=zeros(1,5)
```

```
a =
```

```
0 0 0 0 0
```

Παραδείγματα (5)

```
>> x1=600;  
>> x2=900;  
>> Rphoto=(x2-x1)*rand(8,1)+x1
```

Rphoto =

1.0e+02 *

```
8.786562418434135  
8.190992588566358  
7.46582692149.737  
7.735575183070316  
6.711850739314565  
7.376546484539793  
8.889265617860739  
7.640417156216904
```



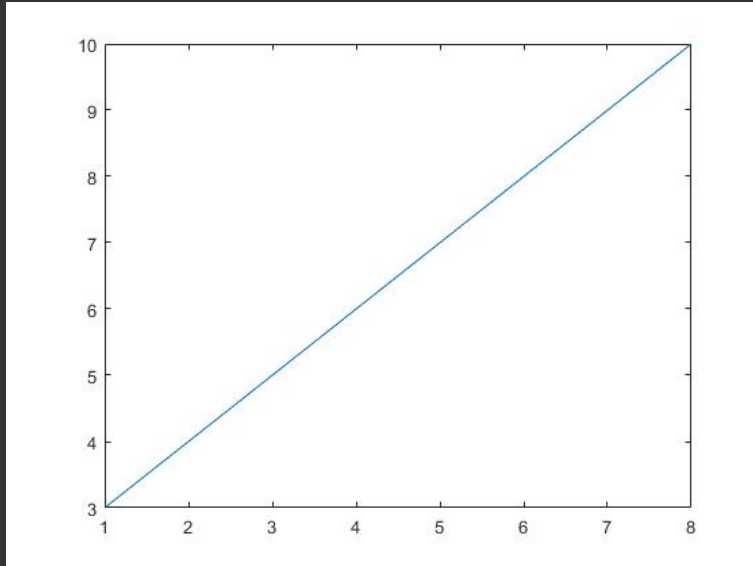
```
>> rstats=datastats(Rphoto)
```

rstats =

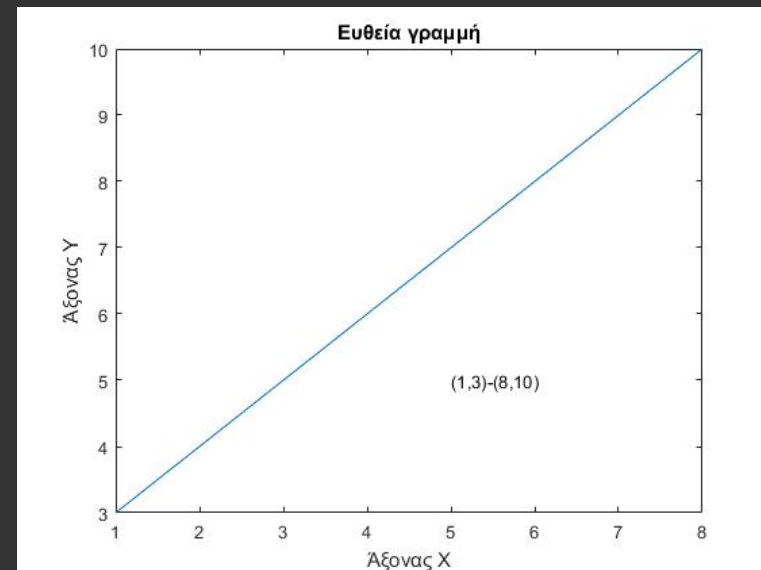
```
num: 8  
max: 8.889265617860739e+02  
min: 6.711850739314565e+02  
mean: 7.849629638676693e+02  
median: 7.68799616964369.e+02  
range: 2.177414878546174e+02  
std: 73.656014869881560
```

Σχεδίαση γραφημάτων (1)

```
>> plot ([1 8],[3 10])
```

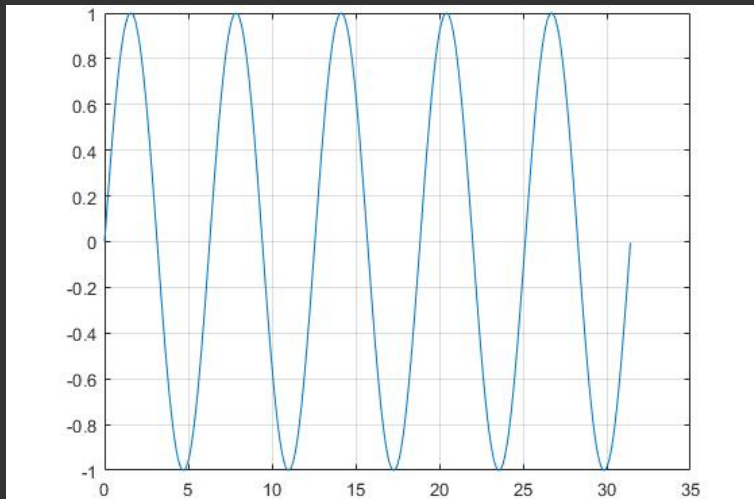


```
>> title('Ευθεία γραμμή')  
>> xlabel('Άξονας X')  
>> ylabel('Άξονας Y')  
>> text(5,5,'(1,3)-(8,10)')
```



Σχεδίαση γραφημάτων (2)

```
>> x=[0:0.01:10*pi];  
>> plot(x,sin(x))  
>> grid on
```



`axis([xmin, xmax, ymin, ymax])`

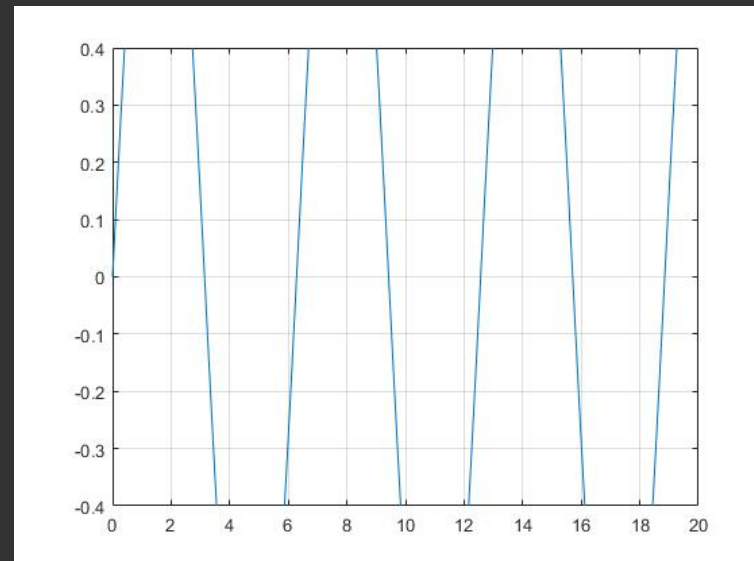
xmin Ελάχιστη τιμή X άξονα

xmax Μέγιστη τιμή X άξονα

ymin Ελάχιστη τιμή Y άξονα

ymax Μέγιστη τιμή Y άξονα

`axis([0,20,-0.4,0.4])`



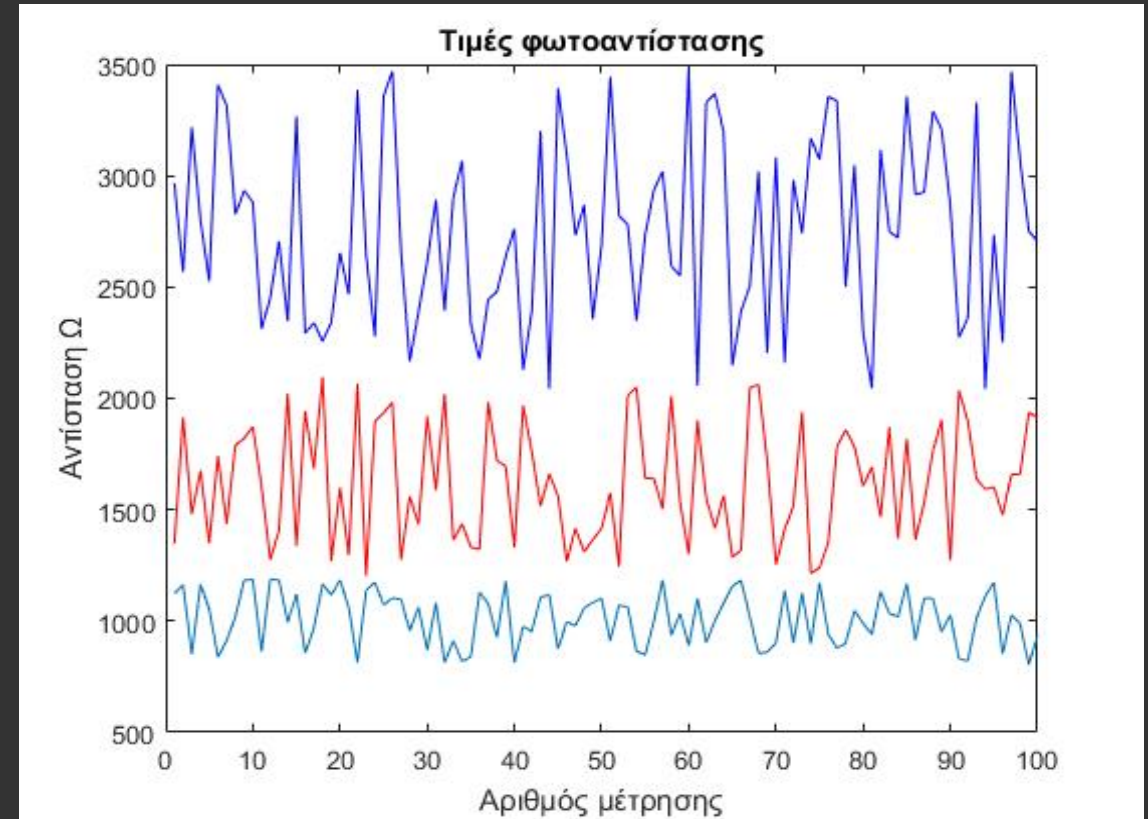
Σχεδίαση γραφημάτων (3Α)

1

```
>> s11=800; s12=1200;  
>> s21=1200; s22=2100;  
>> s31=2000; s32=3500;  
>> r1=(s12-s11)*rand(100,1)+s11;  
>> r2=(s22-s21)*rand(100,1)+s21;  
>> r3=(s32-s31)*rand(100,1)+s31;
```

2

```
>> plot(r1)  
>> hold  
Current plot held  
>> plot(r2,'r')  
>> plot(r3,'b')  
>> title('Τιμές φωτοαντίστασης')  
>> xlabel('Αριθμός μέτρησης')  
>> ylabel('Αντίσταση Ω')
```



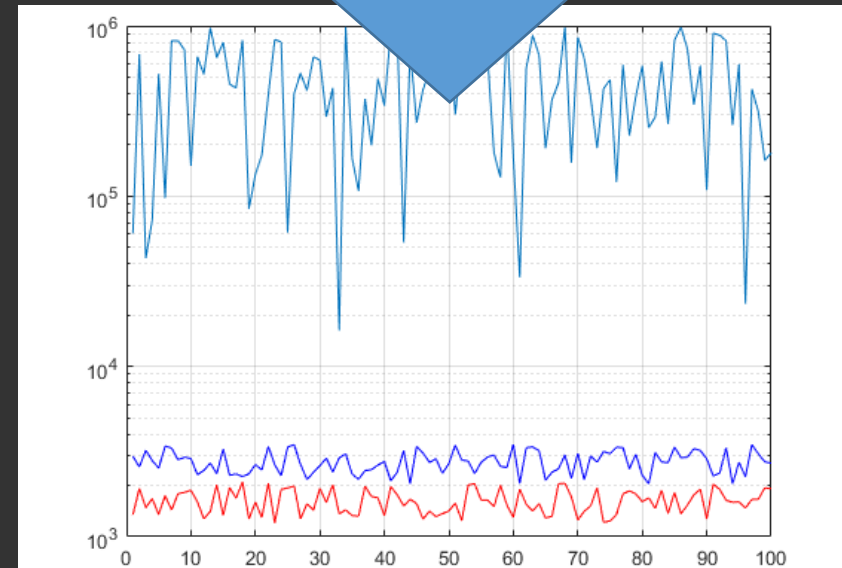
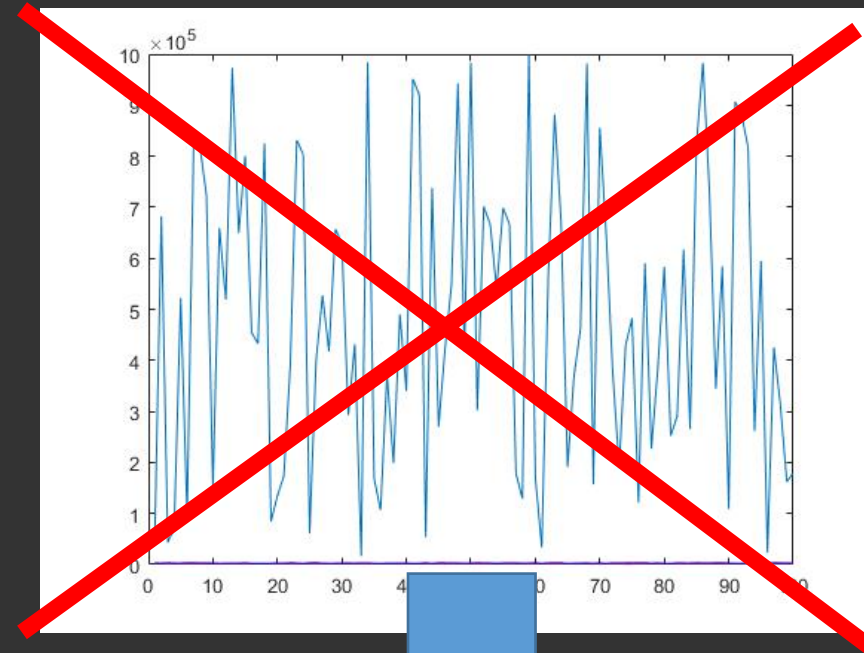
Σχεδίαση γραφημάτων (3B)

3

```
>> s12=1000000;  
>> r1=(s12-s11)*rand(100,1)+s11;
```

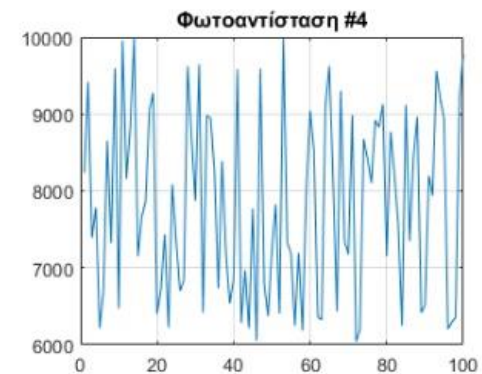
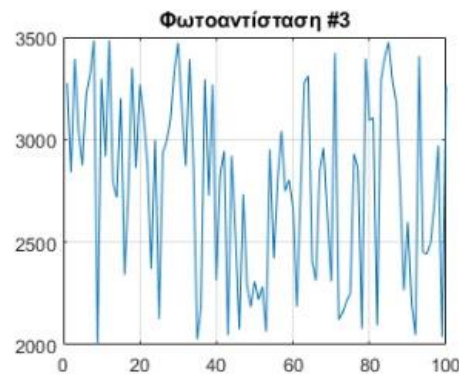
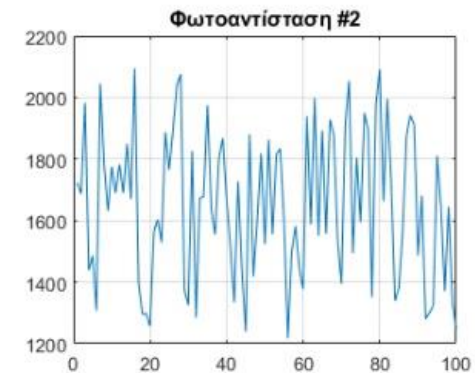
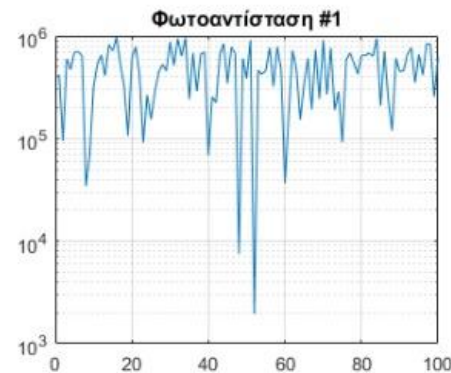
4

```
>> figure  
>> semilogy(r1)  
>> grid on  
>> hold;  
>> semilogy(r2,'r')  
>> semilogy(r3,'b')
```



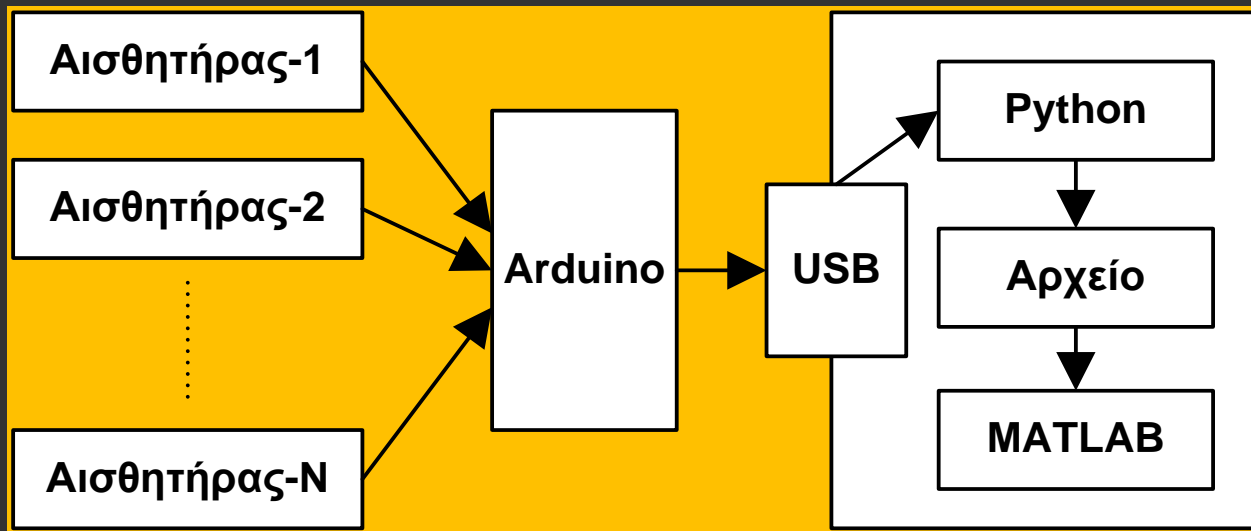
Σχεδίαση γραφημάτων (4)

```
>> subplot(2,2,1); semilogy(r1);  
title('Φωτοαντίσταση #1');  
>> grid on  
>> subplot(2,2,2); plot(r2);  
title('Φωτοαντίσταση #2');  
>> grid on  
>> subplot(2,2,3); plot(r3);  
title('Φωτοαντίσταση #3');  
>> grid on  
>> subplot(2,2,4); plot(r4);  
title('Φωτοαντίσταση #4');  
>> grid on
```



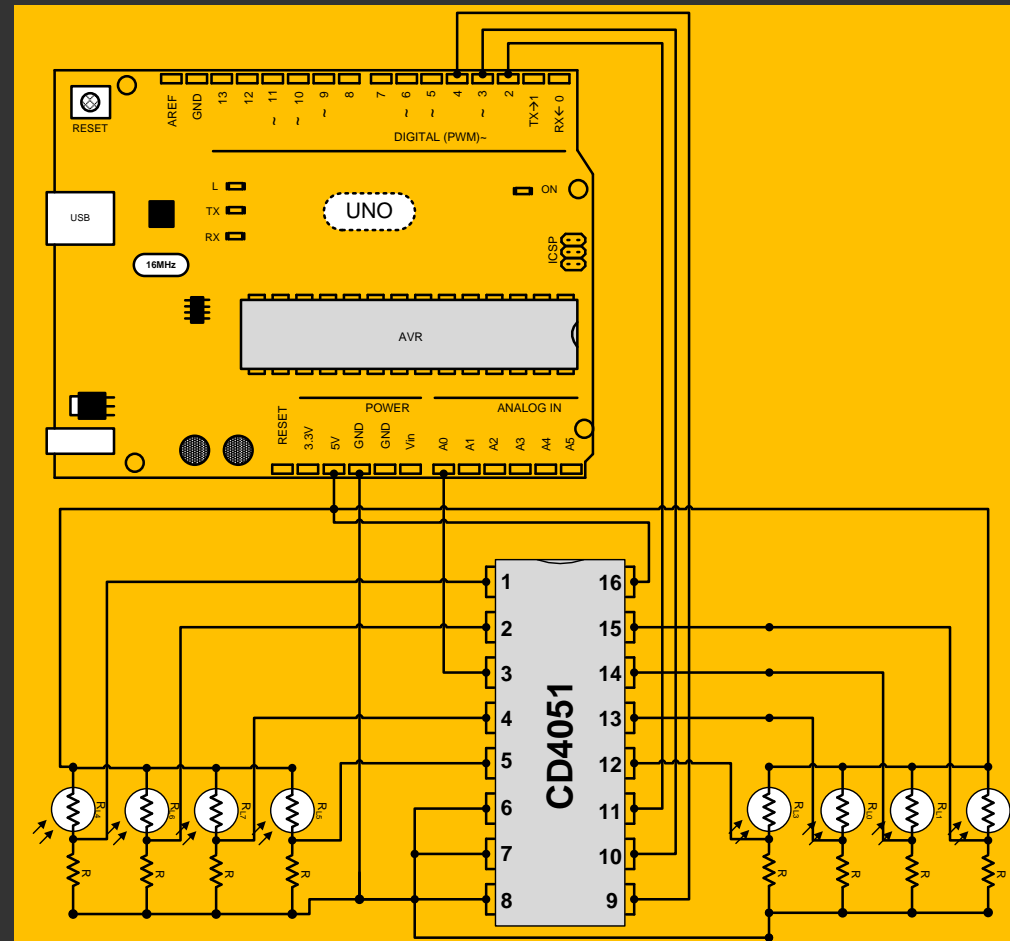
Arduino-Python-MATLAB (1)

Σενάριο



Arduino-Python-MATLAB (2)

Πειραματική διάταξη: Οκτώ αισθητήρες
με αναλογική πολυπλεξία



Arduino-Python-MATLAB (3)

```
byte pin[3]={4,3,2};
void setup()
{
  Serial.begin(9600);
  for(byte i=0;i<3;i++)
    pinMode(pin[i],OUTPUT);
}

void loop()
{
  byte bit3[3]={0,0,0};

  float values[8];
  for(byte m=0;m<8;m++)
  {
    if (m>0)
    {
      if ((m%2)==0) bit3[1]=!bit3[1];
      if ((m%4)==0) bit3[0]=!bit3[0];
    }
  }
```

Arduino

```
for(byte i=0;i<3;i++)
  digitalWrite(pin[i],bit3[i]);
int analog=analogRead(0);
float V0=(analog*4.88)/1000;
float I=V0/10000;
float Ima=I*1000;
float VL=5-V0;
float RL=VL/I;
values[m]=RL;

bit3[2]=!bit3[2];
}

for(byte i=0;i<8;i++)
{if (values[i]<1000) Serial.print('0');
Serial.print(values[i],3);
Serial.print(",");
}
Serial.println("");
delay(100);
}
```

Arduino-Python-MATLAB (4)

```
import serial
import csv
ser = serial.Serial('COM3')
n=1000
i=0
math=[0]*n
while (i<n):
    ser_bytes = str(ser.readline())
    wline2=ser_bytes[2:73]
    math[i]=wline2
    i=i+1

csvfile=open('e:\presistors.csv','w',newline='')
f=csv.writer(csvfile,delimiter="\t",quoting=csv.QUOTE_NON
E)

for row in math:
    f.writerow([row])
csvfile.close()
```

Python

Arduino-Python-MATLAB (5)

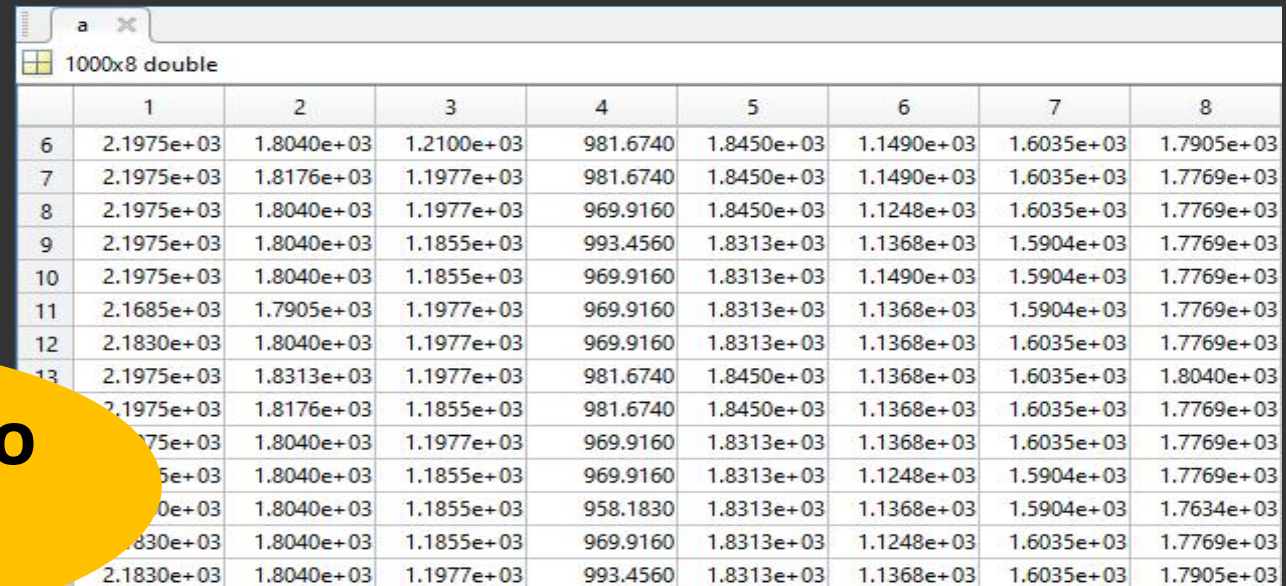
**Άνοιγμα στο
Excel
Επιβεβαίωση**

	A	B	C	D	E	F	G	H
1	2197.501	1817.649	1197.706	981.674	1831.295	1136.849	1603.512	1775.693
2	2212.039	1804.033	1185.482	969.916	1831.295	1136.849	1590.386	1776.898
3	2197.501	1804.033	1185.482	969.916	1831.295	1136.849	1590.386	1776.898
4	2182.998	1804.033	1185.482	993.456	1831.295	1136.849	1603.512	1776.898
5	2182.998	1804.033	1185.482	969.916	1831.295	1136.849	1603.512	1776.898
6	2197.501	1804.033	1209.957	981.674	1844.973	1148.968	1603.512	1790.45
7	2197.501	1817.649	1197.706	981.674	1844.973	1148.968	1603.512	1776.898
8	2197.501	1804.033	1197.706	969.916	1844.973	1124.757	1603.512	1776.898
9	2197.501	1804.033	1185.482	993.456	1831.295	1136.849	1590.386	1776.898
10	2197.501	1804.033	1185.482	969.916	1831.295	1148.968	1590.386	1776.898
11	2168.529	1790.45	1197.706	969.916	1831.295	1136.849	1590.386	1776.898
12	2182.998	1804.033	1197.706	969.916	1831.295	1136.849	1603.512	1776.898
13	2197.501	1831.295	1197.706	981.674	1844.973	1136.849	1603.512	1804.033
14	1817.649	1185.482	981.674	1844.973	1136.849	1603.512	1776.898	
15	1804.033	1197.706	969.916	1831.295	1136.849	1603.512	1776.898	
16	1804.033	1185.482	969.916	1831.295	1124.757	1590.386	1776.898	
17	1804.033	1185.482	958.183	1831.295	1136.849	1590.386	1763.377	
18	2182.998	1804.033	1185.482	969.916	1831.295	1124.757	1603.512	1776.898
19	2182.998	1804.033	1197.706	993.456	1831.295	1136.849	1603.512	1790.45
20	2212.039	1804.033	1197.706	981.674	1831.295	1148.968	1603.512	1776.898
21	2197.501	1804.033	1197.706	981.674	1831.295	1136.849	1603.512	1763.377
22	2197.501	1804.033	1197.706	969.916	1831.295	1148.968	1590.386	1776.898
23	2182.998	1804.033	1185.482	969.916	1831.295	1136.849	1590.386	1776.898
24	2182.998	1804.033	1185.482	969.916	1831.295	1136.849	1590.386	1776.898
25	2212.039	1804.033	1197.706	969.916	1831.295	1136.849	1603.512	1776.898
26	2197.501	1790.45	1197.706	981.674	1844.973	1148.968	1603.512	1790.45
27	2197.501	1817.649	1197.706	981.674	1844.973	1161.112	1603.512	1790.45
28	2197.501	1804.033	1209.957	969.916	1844.973	1136.849	1603.512	1790.45
29	2197.501	1804.033	1185.482	969.916	1831.295	1136.849	1590.386	1776.898
30	2197.501	1804.033	1185.482	969.916	1831.295	1136.849	1603.512	1776.898

Arduino-Python-MATLAB (6)

```
a=csvread('e:\presistors.csv');
```

Άνοιγμα στο
MATLAB



	1	2	3	4	5	6	7	8
6	2.1975e+03	1.8040e+03	1.2100e+03	981.6740	1.8450e+03	1.1490e+03	1.6035e+03	1.7905e+03
7	2.1975e+03	1.8176e+03	1.1977e+03	981.6740	1.8450e+03	1.1490e+03	1.6035e+03	1.7769e+03
8	2.1975e+03	1.8040e+03	1.1977e+03	969.9160	1.8450e+03	1.1248e+03	1.6035e+03	1.7769e+03
9	2.1975e+03	1.8040e+03	1.1855e+03	993.4560	1.8313e+03	1.1368e+03	1.5904e+03	1.7769e+03
10	2.1975e+03	1.8040e+03	1.1855e+03	969.9160	1.8313e+03	1.1490e+03	1.5904e+03	1.7769e+03
11	2.1685e+03	1.7905e+03	1.1977e+03	969.9160	1.8313e+03	1.1368e+03	1.5904e+03	1.7769e+03
12	2.1830e+03	1.8040e+03	1.1977e+03	969.9160	1.8313e+03	1.1368e+03	1.6035e+03	1.7769e+03
13	2.1975e+03	1.8313e+03	1.1977e+03	981.6740	1.8450e+03	1.1368e+03	1.6035e+03	1.8040e+03
14	2.1975e+03	1.8176e+03	1.1855e+03	981.6740	1.8450e+03	1.1368e+03	1.6035e+03	1.7769e+03
15	2.1975e+03	1.8040e+03	1.1977e+03	969.9160	1.8313e+03	1.1368e+03	1.6035e+03	1.7769e+03
16	2.1975e+03	1.8040e+03	1.1855e+03	969.9160	1.8313e+03	1.1248e+03	1.5904e+03	1.7769e+03
17	2.1975e+03	1.8040e+03	1.1855e+03	958.1830	1.8313e+03	1.1368e+03	1.5904e+03	1.7634e+03
18	2.1830e+03	1.8040e+03	1.1855e+03	969.9160	1.8313e+03	1.1248e+03	1.6035e+03	1.7769e+03
19	2.1830e+03	1.8040e+03	1.1977e+03	993.4560	1.8313e+03	1.1368e+03	1.6035e+03	1.7905e+03

Arduino-Python-MATLAB (7)

```
>> datastats(a(:,1))
```

```
ans =
```

```
num: 9.00  
max: 2.2266e+03  
min: 2.1685e+03  
mean: 2.1964e+03  
median: 2.1975e+03  
range: 58.0830  
std: 9.3596
```

**Βασικοί &
αυτόματοι
υπολογισμοί
MATLAB**

```
clc  
for i=1:1:8  
    fprintf('Sensor %d',i)  
    datastats(a(:,i))  
end
```

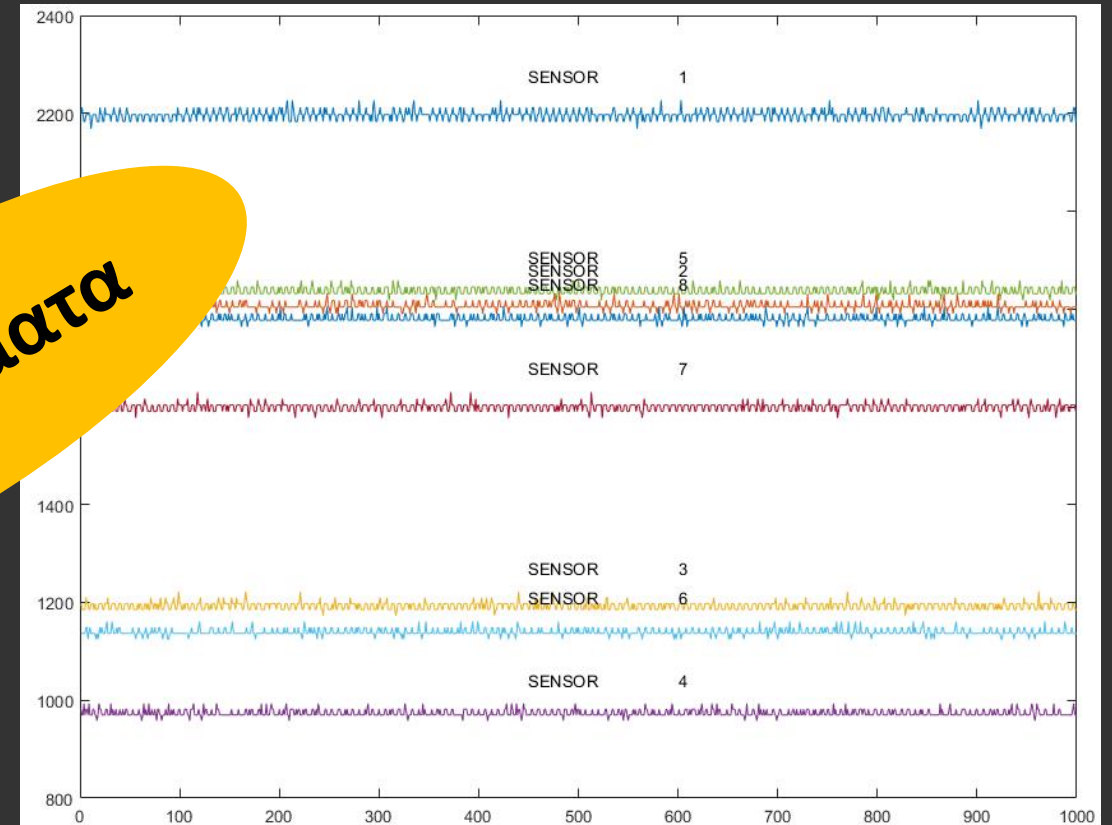


Arduino-Python-MATLAB (8)

```
clc
close all;
samples=1000;
middle=(samples/2)-50;
number=middle+150;
plot(a(:,1))
hold;
for i=2:1:8
    plot(a(:,i))
end

text1='SENSOR ';
for i=1:1:8
    text(middle,max(a(:,i))+50,text1)
    text(number,max(a(:,i))+50,num2str(i))
end
```

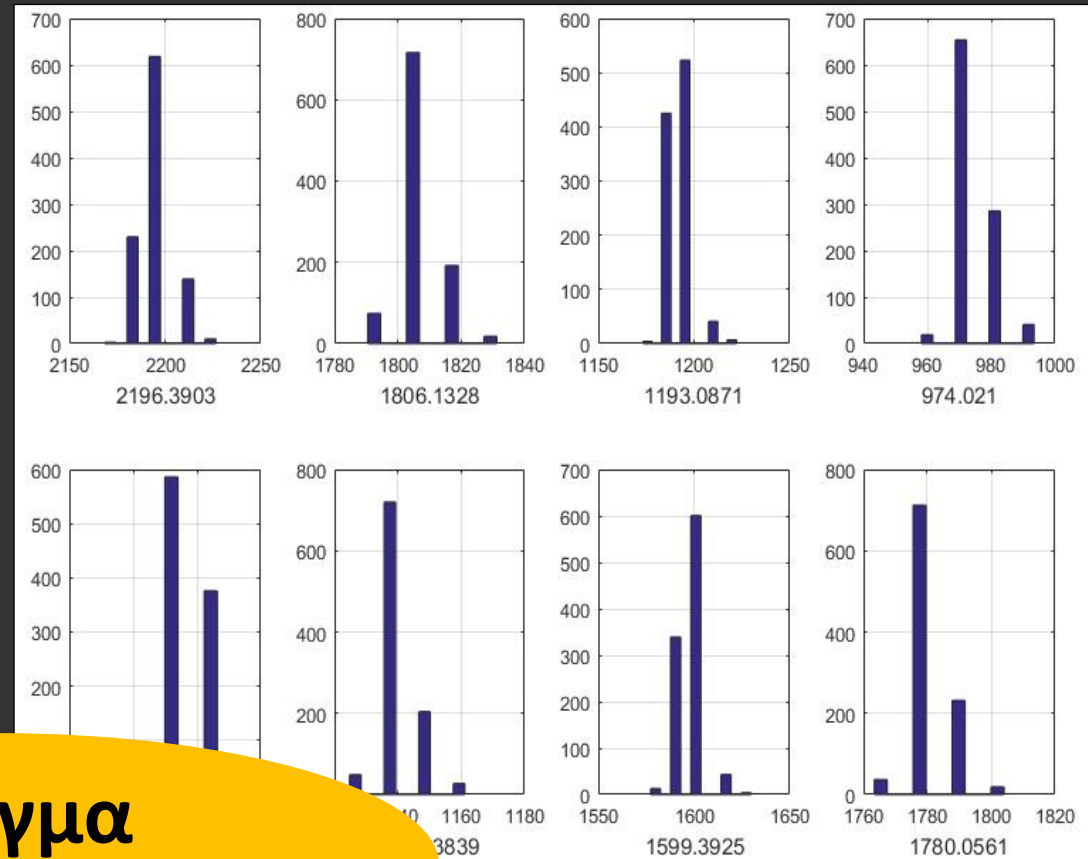
Διαγράμματα



Arduino-Python-MATLAB (9)

```
clc
close all;
for i=1:1:8
    subplot(2,4,i); hist(a(:,i)); grid on;
    m(i)=mean(a(:,i)); ms=num2str(m(i));
    stdi(i)=std(a(:,i));
    xlabel(ms);
end
figure;
subplot(1,2,1); bar(stdi); xlabel('STD');
subplot(1,2,2); bar(m); xlabel('MEAN');

mean(m)
```



**Πλέγμα
διαγραμμάτων**