Σχολή
Ηλεκτρολόγων
Μηχανικών &
Μηχανικών
Υπολογιστών

ΗΛΕΚΤΡΟΝΙΚΑ ΙΣΧΥΟΣ

1η Εργαστηριακή Άσκηση

Εξοικείωση με τα απαραίτητα εργαλεία του Matlab

Περιεχόμενα

Μετασχηματισμός Fourier

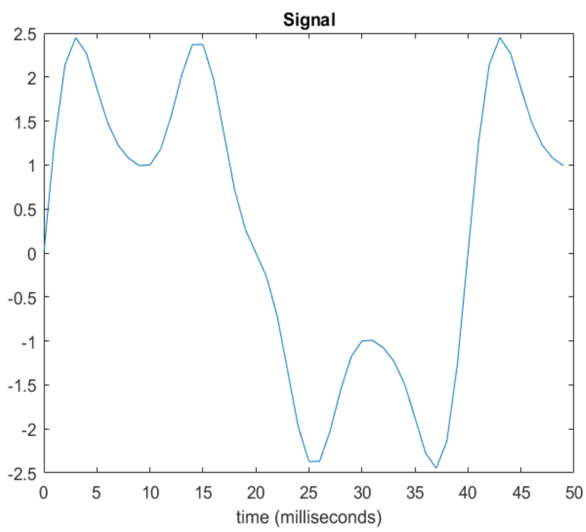
Στα ηλεκτρονικά κυκλώματα ισχύος οι κυματομορφές εξόδου, εκτός από τον κύριο όρο στη θεμελιώδη συχνότητα, περιέχουν και όρους σε ανεπιθύμητες συχνότητες που είναι αρμονικές (πολλαπλάσια) της θεμελιώδους συχνότητας. Οι όροι αυτοί υπολογίζονται μέσω της ανάλυσης Fourier , το οποίο έχει μεγάλη σημασία, καθώς δίνεται η δυνατότητα του περιορισμού των αρμονικών συνιστωσών με τη χρήση κατάλληλων φίλτρων.

Παρακάτω δίνονται παραδείγματα κώδικα Matlab, όπου γίνεται ο μετασχηματισμός Fourier των σημάτων με χρήση της συνάρτησης $\boldsymbol{fft}$.

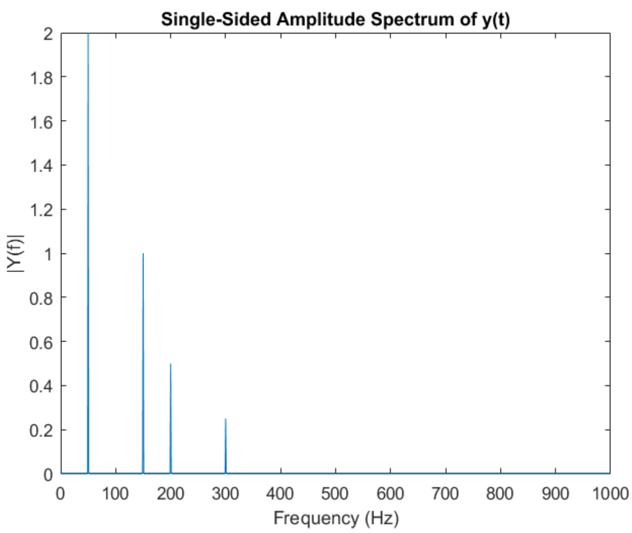
1ο Παράδειγμα

```
Fs = 2000;                % Sampling frequency
Ts = 1/Fs;                % Sample time
L = 2000                  % Length of signal
t = (0:L-1)*Ts;           % Time vector

% Signal to apply Fourier transformation
y = 2*sin(2*pi*50*t+sin(2*pi*150*t+0.5*sin(2*pi*200*t)+0.25*sin(2*pi*300*t);

% Plot Signal
figure
plot(Fs*t(1:50),y(1:50))
title('Signal')
xlabel('time (milliseconds)')

% Fourier transformation
NFFT = 2000;
Y = fft(y,NFFT)/L;
f = Fs/2*linspace(0,1,NFFT/2+1);

% Plot single-sided amplitude spectrum.
figure
plot(f,2*abs(Y(1:NFFT/2+1)))
title('Single-Sided Amplitude Spectrum of y(t)')
xlabel('Frequency (Hz)')
ylabel('|Y(f)|')
```

Το εξεταζόμενο σήμα και το προκύπτον φασματικό του περιεχόμενο δίνονται στα Σχήματα 1.Α, 1.Β, αντίστοιχα.



A)



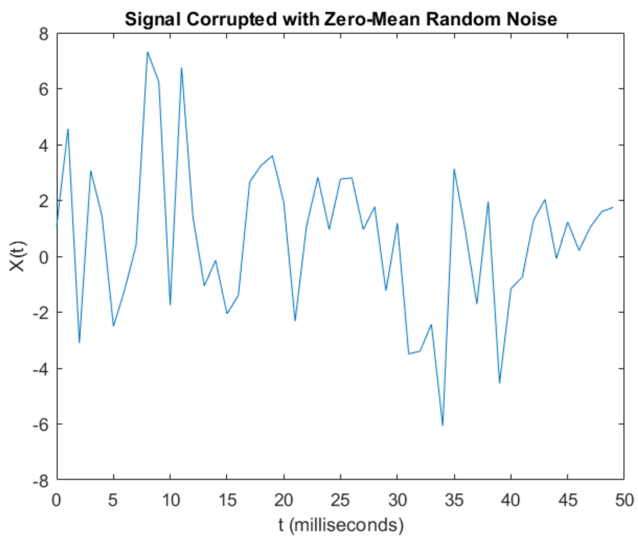
B)

Σχήμα 1. Αποτελέσματα του παραδείγματος κώδικα Matlab για το μετασχηματισμό Fourier.

2ο Παράδειγμα

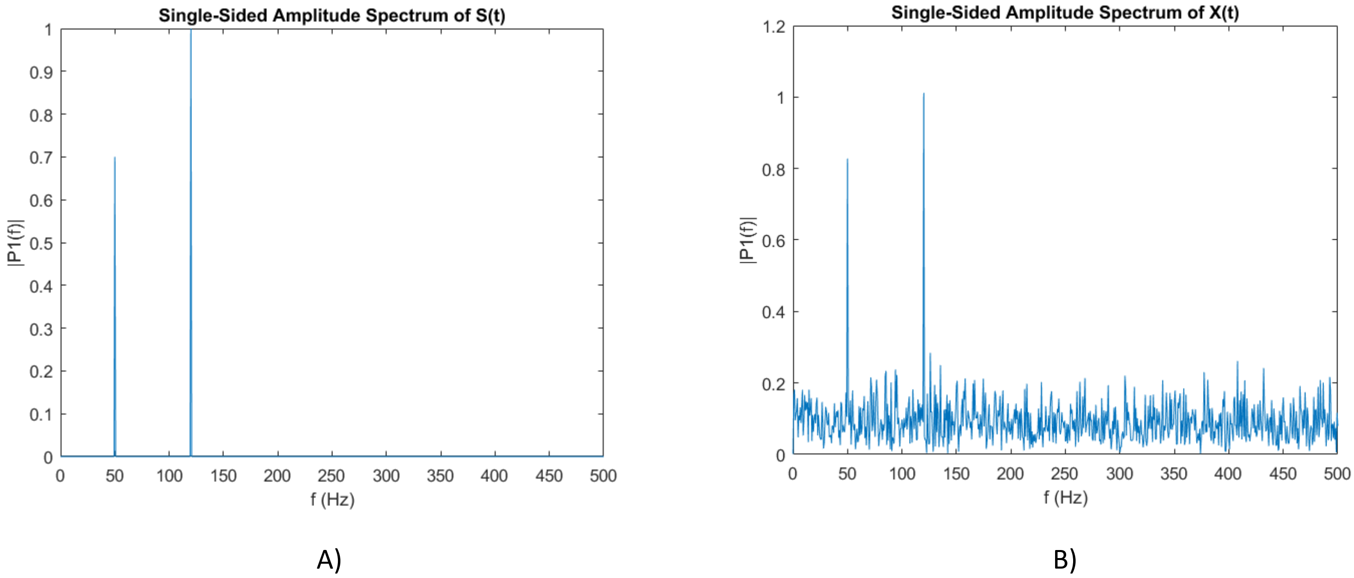
```
Fs = 1000;            % Sampling frequency
T = 1/Fs;             % Sampling period
L = 1500;             % Length of signal
t = (0:L-1)*T;        % Time vector

S = 0.7*sin(2*pi*50*t) + sin(2*pi*120*t);

f = Fs*(0:(L/2))/L;

% Fourier transform of the original, uncorrupted signal
Y = fft(S);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);

figure
plot(f,P1)
title('Single-Sided Amplitude Spectrum of S(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')

% Corrupt the signal with zero-mean white noise with a variance of 4.
X = S + 2*randn(size(t));

% Plot the noisy signal in the time domain
figure
plot(1000*t(1:50),X(1:50))
title('Signal Corrupted with Zero-Mean Random Noise')
xlabel('t (milliseconds)')
ylabel('X(t)')

% Compute the Fourier transform of the signal
Y = fft(X);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);

figure
plot(f,P1)
title('Single-Sided Amplitude Spectrum of X(t)')
xlabel('f (Hz)')
ylabel('|P1(f)|')
```

Το εξεταζόμενο σήμα που προέκυψε με τη προσθήκη λευκού θορύβου, δίνεται στο Σχήμα 2.



Σχήμα 2. Σήμα με λευκό Gaussian θόρυβο

Το φασματικό περιεχόμενο του σήματος πριν και μετά την προσθήκη θορύβου δίνονται στα Σχήματα 3.A, 3.B, αντίστοιχα.

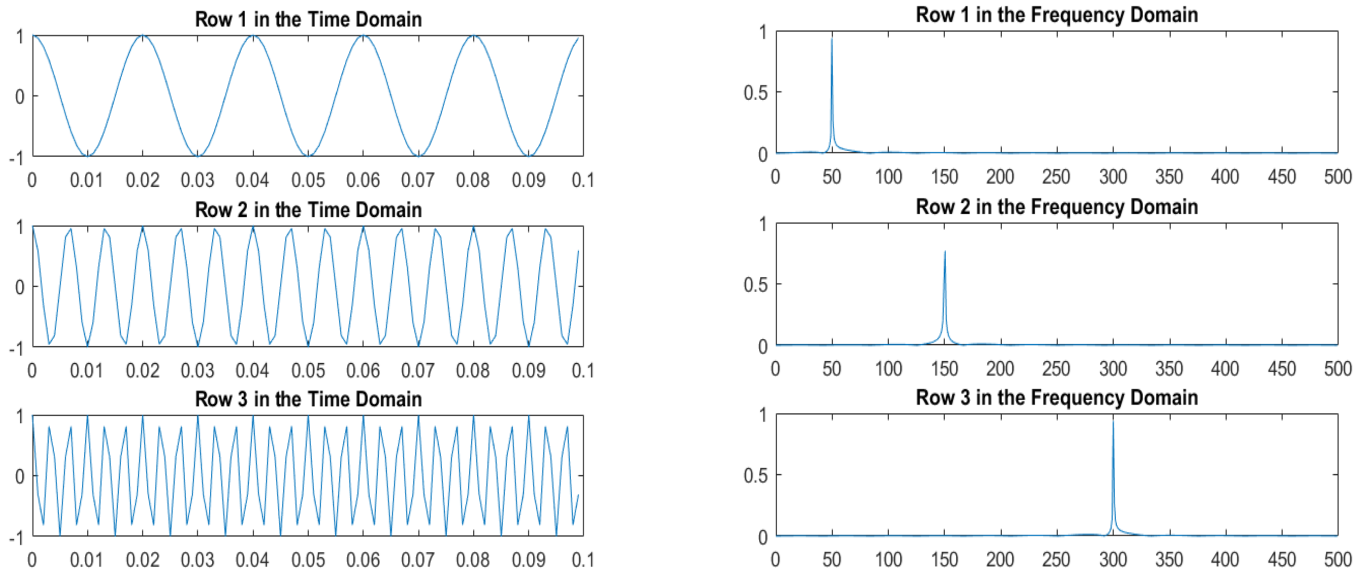


Σχήμα 3. Αποτελέσματα του παραδείγματος κώδικα Matlab για το μετασχηματισμό Fourier.

3º Παράδειγμα

```
Fs = 1000;                    % Sampling frequency
T = 1/Fs;                     % Sampling period
L = 1000;                     % Length of signal
t = (0:L-1)*T;                % Time vector

x1 = cos(2*pi*50*t);          % First row wave
x2 = cos(2*pi*150*t);         % Second row wave
x3 = cos(2*pi*300*t);         % Third row wave

X = [x1; x2; x3];

figure
for i = 1:3
    subplot(3,1,i)
    plot(t(1:100),X(i,1:100))
    title(['Row ',num2str(i),' in the Time Domain'])
end

n = 2^nextpow2(L);
dim = 2;

Y = fft(X,n,dim);
P2 = abs(Y/L);
P1 = P2(:,1:n/2+1);
P1(:,2:end-1) = 2*P1(:,2:end-1);

figure
for i=1:3
    subplot(3,1,i)
    plot(0:(Fs/n):(Fs/2-Fs/n),P1(i,1:n/2))
    title(['Row ',num2str(i),' in the Frequency Domain'])
end
```

Παρακάτω δίνονται τα 100 πρώτα στοιχεία από κάθε γραμμή του πίνακα X κατά σειρά, καθώς και το προκύπτον φασματικό περιεχόμενο τους στα Σχήματα 4.A, 4.B, αντίστοιχα.



A) B)

Σχήμα 4. Αποτελέσματα του παραδείγματος κώδικα Matlab για το μετασχηματισμό Fourier.

Συστήματα στο χώρο κατάστασης - Εντολές ss και c2d

Με τη συνάρτηση $\boldsymbol{ss}$ του Matlab ορίζουμε το μοντέλο του συστήματος στην παρακάτω μορφή,

$$\frac{d}{dt}x = Ax + Bu$$

$$y = Cx + Du$$

και χρησιμοποιώντας τη συνάρτηση $\boldsymbol{c2d}$ το εν λόγω σύστημα μετατρέπεται σε σύστημα διακριτού χρόνου, με χρόνο δειγματοληψίας T_s, της μορφής,

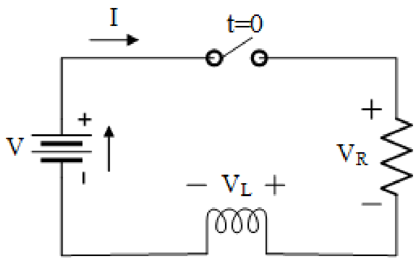
$$x\big((k+1)\cdot T_s\big) = \ A_d x(k\cdot T_s) + B_d u(k\cdot T_s)$$

$$y(k\cdot T_s) = C_d x(k\cdot T_s) + D_d u(k\cdot T_s)$$

Παράδειγμα εφαρμογής των ss και c2d

```
% Ορισμός του συστήματος στον χώρο κατάστασης

A=[0 1;-2 -1]; B=[0;0.3]; C=[1 1;0 -0.3]; D=0;

sys=ss(A,B,C,D);

% Μετατροπή του συστήματος συνεχούς χρόνου σε διακριτό με Ts=0.1s

sysd=c2d(sys,0.1);


step(sys)
hold
step(sysd,'r')
```

Εξίσωση κατάστασης RL φορτίου



Σχήμα 5. RL κύκλωμα

Η διαφορική εξίσωση του RL φορτίου είναι η εξής:

$$V = R \cdot I_L + L \cdot \frac{dI_L}{dt}$$

Προκειμένου να φέρουμε την παραπάνω εξίσωση σε μορφή εξισώσεων κατάστασης, λύνουμε ως προς την παράγωγο, όπως φαίνεται παρακάτω,

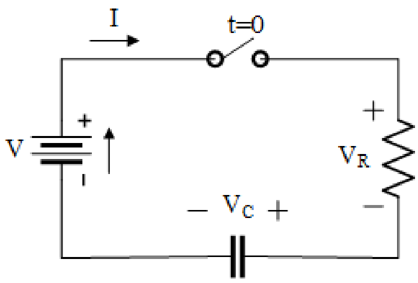
$$\frac{dI_L}{dt} = -\frac{R}{L} \cdot I_L + \frac{1}{L} \cdot V_L$$

και έστω ότι θεωρούμε ως έξοδο το ρεύμα I_L, δηλαδή $y = I_L$, προκύπτουν οι πίνακες,

$$A = -\frac{R}{L} \quad B = \frac{1}{L} \quad C = 1 \quad D = 0$$

Οι παραπάνω πίνακες A, B, C, D ορίζουν το μοντέλο ενός RL φορτίου στο χώρο κατάστασης.

Εξίσωση κατάστασης RC φορτίου



Σχήμα 6. RC κύκλωμα

Η διαφορική εξίσωση του RC φορτίου είναι η εξής:

$$\frac{V_C}{R} + C \cdot \frac{dV_C}{dt} = \frac{V}{R}$$

Προκειμένου να φέρουμε την παραπάνω εξίσωση σε μορφή εξισώσεων κατάστασης, λύνουμε ως προς την παράγωγο, όπως φαίνεται παρακάτω,

$$\frac{dV_C}{dt} = -\frac{1}{RC} \cdot V_C + \frac{1}{RC} \cdot V$$

και έστω ότι θεωρούμε ως έξοδο το ρεύμα V_C, δηλαδή $y = V_C$, προκύπτουν οι πίνακες,

$$A = -\frac{1}{RC} \quad B = \frac{1}{RC} \quad C = 1 \quad D = 0$$

Οι παραπάνω πίνακες A, B, C, D ορίζουν το μοντέλο ενός RC φορτίου στο χώρο κατάστασης.

Εξισώσεις διαφορών διακριτού χρόνου

Η διαφορική εξίσωση του RC φορτίου,

$$\frac{V_C(t)}{R} + C \cdot \frac{dV_C(t)}{dt} = \frac{V(t)}{R}$$

μετατρέπεται σε εξίσωση διαφορών διακριτού χρόνου, ως εξής:

$$\frac{V_C(k \cdot \Delta t)}{R} + C \cdot (V_C(k \cdot \Delta t) - V_C((k-1) \cdot \Delta t) = \frac{V(k \cdot \Delta t)}{R}$$

Λύνοντας ως $V_C(k \cdot \Delta t)$ προκύπτει,

$$V_C(k \cdot \Delta t) = \frac{RC}{1+RC} V_C\big((k-1) \cdot \Delta t\big) + \frac{1}{1+RC} V(k \cdot \Delta t)$$

Συναρτήσεις μεταφοράς

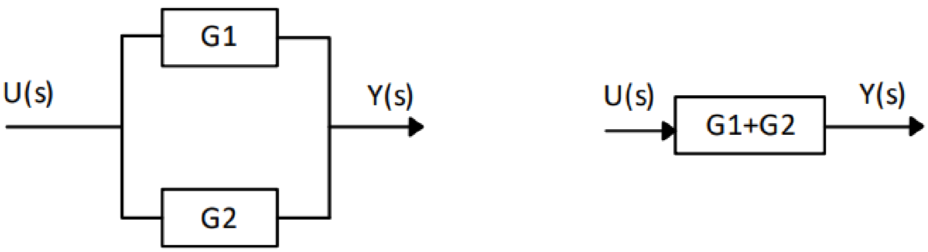
Οι συναρτήσεις μεταφοράς είναι μία αναπαράσταση στο πεδίο της συχνότητας των γραμμικών χρονικά αμετάβλητων συστημάτων. Προσδιορίζονται από τα πολυώνυμα $A(s)$ και $B(s)$ του αριθμητή και του παρονομαστή τους, όπως φαίνεται παρακάτω.

$$H(s) = \frac{A(s)}{B(s)} = \frac{a_1 s^n + a_2 s^{n-1} + \cdots + a_{n+1}}{b_1 s^m + b_2 s^{m-1} + \cdots + b_{m+1}}$$

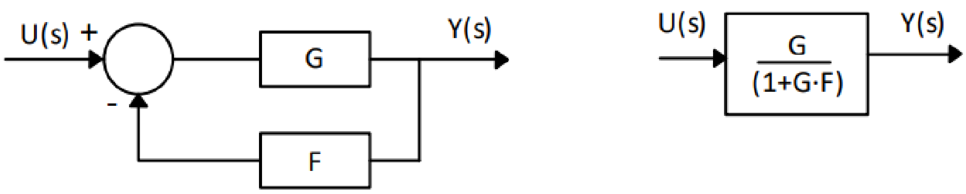
Απλοποίηση συναρτήσεων μεταφοράς

Σύνδεση σε σειρά

Σύνδεση παράλληλα



Σύνδεση με ανάδραση



Εντολή tf

Με τη συνάρτηση $\boldsymbol{tf}$ του Matlab δημιουργείται ένα μοντέλο συνάρτησης μεταφοράς συνεχούς χρόνου, ορίζοντας κατάλληλα τους συντελεστές του αριθμητή και του παρονομαστή.

Για παράδειγμα, η συνάρτηση μεταφοράς $\mathrm{H}(s) = \frac{s+2}{s^2+4s+5}$ ορίζεται όπως φαίνεται παρακάτω.

```
H = tf([1 2],[1 4 5]);
```

Εντολή zpk

Τα μοντέλα $ZPK(zero - pole - gain)$ είναι η παραγοντική μορφή των συναρτήσεων μεταφοράς, όπως φαίνεται παρακάτω και ορίζονται μέσω της συνάρτησης $\boldsymbol{zpk}$.

$$H(s) = K\frac{(s - z_1) \dots (s - z_n)}{(s - p_1) \dots (s - p_m)}$$

1ο παράδειγμα

$$H(s) = \frac{-2s}{(s-2)(s^2 - 2s + 2)}$$

```
H = zpk(0,[2 1+1i 1-1i],-2);
```

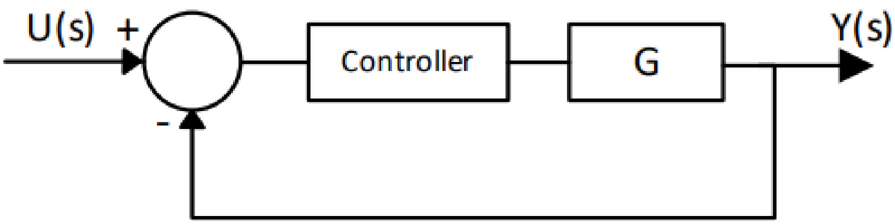
2ο παράδειγμα

$$H(s) = \frac{1}{(s+1)^2}$$

```
H = zpk([],[-1 -1],1);
```

Εντολή pidtune

Η εντολή $\boldsymbol{pidtune}$ ρυθμίζει βέλτιστα τις παραμέτρους του ελεγκτή και εφαρμόζεται σε συστήματα με ανάδραση της εξής μορφής,



Παράδειγμα σχεδίασης PI ελεγκτή συστήματος με συνάρτηση μεταφοράς G

Έστω ελεγχόμενο σύστημα με συνάρτηση μεταφοράς $G = \frac{1}{(s+1)^2}$, τότε η σχεδίαση ενός PI ελεγκτή υλοποιείται σε κώδικα Matlab ως εξής.

```
G = zpk([],[-1 -1],1);
C1 = pidtune(G,'PI');
```

Εντολή pidTuner

Με χρήση της εντολής $\boldsymbol{pidTuner(G, type)}$ ρυθμίζονται αυτόματα και βέλτιστα οι παράμετροι του ελεγκτή, ώστε να επιτευχθεί ισορροπία μεταξύ απόδοσης και ευστάθειας. Μέσω του γραφικού περιβάλλοντος μπορεί να εξεταστεί η απόδοση του ελεγκτή στο πεδίο του χρόνου και της συχνότητας.

Παράδειγμα

```
G = zpk([],[-1 -1],1); % Continuous-time zpk model
G_d = c2d(G,0.1);      % Discrete-time zpk model

C = pidTuner(G_d,'PI');
```

Συναρτήσεις interp1 και fsolve

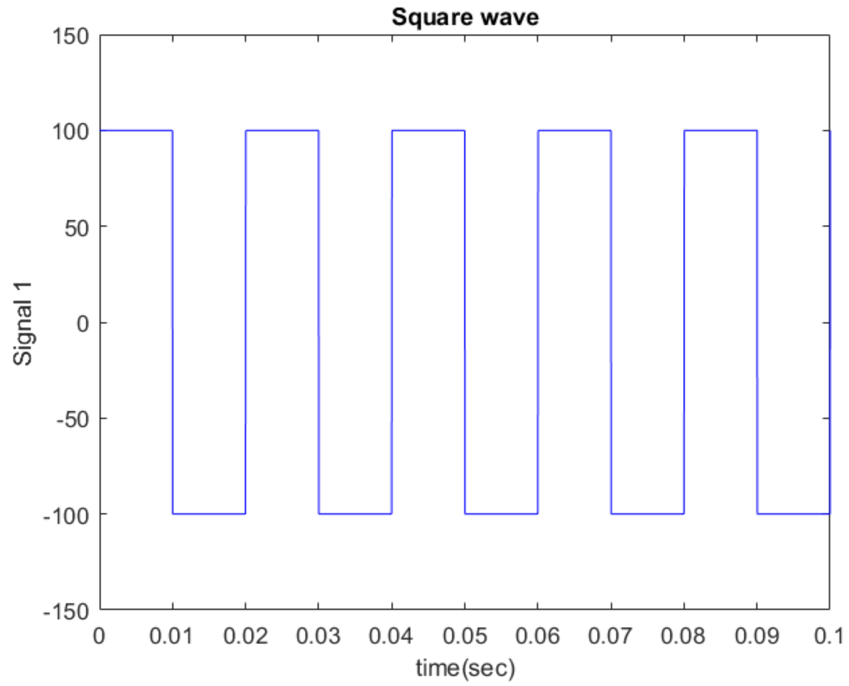
Η συνάρτηση $\boldsymbol{interp1}$ πραγματοποιεί γραμμική παρεμβολή. Τα δεδομένα που χρειάζεται είναι ζεύγη τιμών μεταβλητής(ων) – συνάρτησης ώστε να μπορεί να υπολογίζει την συνάρτηση σε ενδιάμεσες τιμές που καθορίζει ο χρήστης. Η $\boldsymbol{fsolve}$ επιλύει εξισώσεις της μορφής F(x)=0. Ο χρήστης ορίζει την συνάρτηση F και το αρχικό σημείο εκκίνησης του αλγορίθμου επίλυσης.

Παρακάτω δίνονται παραδείγματα κώδικα Matlab εφαρμογής των συναρτήσεων $interp1$ και $fsolve$.

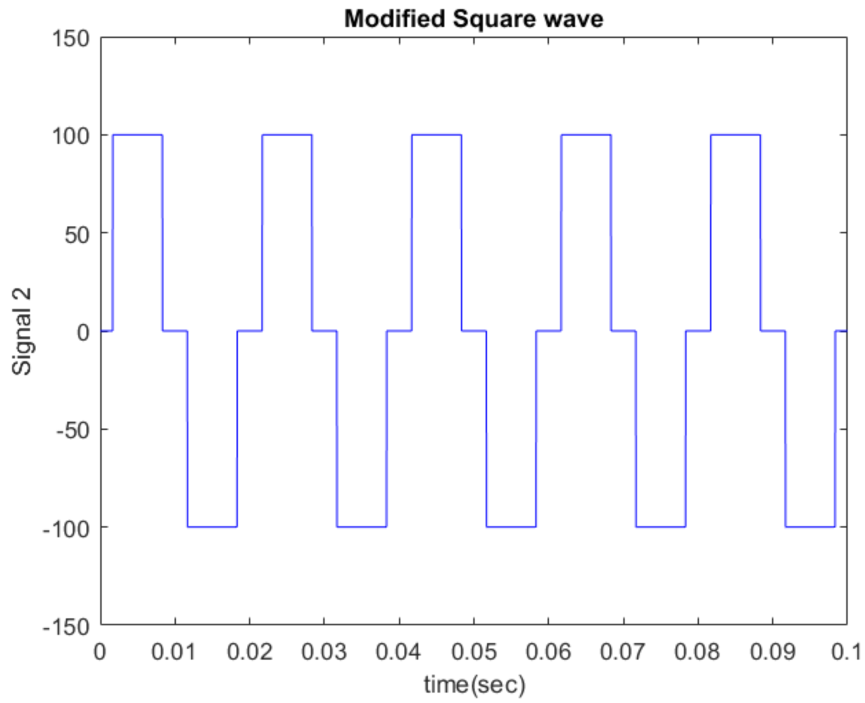
```
% Παράδειγμα εφαρμογής της interp1
xs=0:0.02/16:0.02; % Διάνυσμα τιμών της μεταβλητής εισόδου στα σημεία
δειγματοληψίας

fs=sin(2*pi*50*xs)./(15*xs+0.05); % Διάνυσμα τιμών της συνάρτησης στις
δειγματολειπτούμενες τιμές της μεταβλητής εισόδου

x=0:0.02/36:0.02; % Διάνυσμα τιμών της μεταβλητής εισόδου στα επιθυμητά σημεία

f=interp1(xs,fs,x,'spline'); % Χρήση της interp1 στο διάνυσμα των
επιθυμητών σημείων

figure
hold
plot(xs,fs)
plot(x,f, 'o')

%================================================================
%================================================================

% Παράδειγμα εφαρμογής της fsolve
global A B C D
A=1;
B=-2;
C=0.6;
D=1;
x0 = 0; % Starting point of the algorithm
options = optimoptions('fsolve','Display','iter'); %Option to display
[xval,fval] = fsolve('F' ,x0,options); % Call solver

%================================================================
function Fval = F(x)
global A B C D
Fval = A*x^3 + B*x^2 + C*x +D;
end
%===============================================================
```

Δεδομένα

Οι γραφικές παραστάσεις των προς ανάλυση σημάτων δίνονται στα Σχήματα 7.Α και 7.Β. Τα σήματα των Σχημάτων 7.Α και 7.Β περιέχονται στα αρχεία signal_1.mat και signal_2.mat, αντίστοιχα. Το διάνυσμα του χρόνου περιέχεται στο αρχείο time.mat. Η συχνότητα των σημάτων είναι ίση με 50Hz.



A)

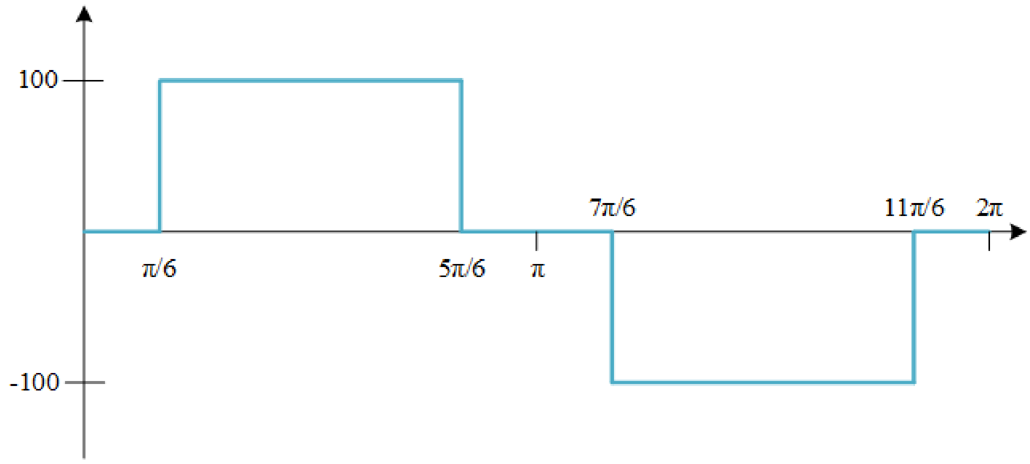


B)

Σχήμα 7. Σήματα προς ανάλυση

Εκτέλεση άσκησης, Παραδοτέα

1. Να χρησιμοποιηθεί η συνάρτηση fft του Matlab για να γίνει ο μετασχηματισμός Fourier των σημάτων των Σχημάτων 7.Α και 7.Β που περιέχονται στα αρχεία signal_1.mat και signal_2.mat, αντίστοιχα. Χρησιμοποιείστε αριθμό σημείων του μετασχηματισμού Fourier, NFFT = 1001000 και μήκος, L = 1001. Ο χρόνος δειγματοληψίας των σημάτων είναι $T_s = 2 \cdot 10^{-5}$ sec.
2. Να δοθούν οι γραφικές παραστάσεις των φασματικών περιεχομένων των σημάτων και να τα σχολιάσετε.
3. Να υπολογίσετε την $3^η$ αρμονική του σήματος που περιέχεται στο αρχείο signal_2.mat και να σχολιάσετε συγκριτικά με το αποτέλεσμα που λάβατε στο $1^ο$ ερώτημα. Παρακάτω δίνεται μία περίοδος του σήματος signal_2.



4. Επαναλάβετε τα βήματα του $1^{ου}$ ερωτήματος, χρησιμοποιώντας αριθμό σημείων μετασχηματισμού Fourier, NFFT = L, όπου L το μήκος των σημάτων που σας δίνονται. Στη συνέχεια, ανακατασκευάστε τα αρχικά σήματα signal_1 και signal_2 στο πεδίο του χρόνου (χωρίς τη χρήση της εντολής ifft του Matlab), λαμβάνοντας υπόψη για το κάθε σήμα,
   α) τις 10 πρώτες αρμονικές
   β) τις 20 πρώτες αρμονικές
   γ) όλες τις αρμονικές
5. Να δοθούν οι γραφικές παραστάσεις των σημάτων που να ανακατασκευάσατε για τις 3 παραπάνω περιπτώσεις και να τα σχολιάσετε.
6. Να δώσετε τον κώδικα Matlab που αναπτύξατε για τα παραπάνω ερωτήματα με τα κατάλληλα σχόλια.