Cepstrum

1. Using Cepstrum to calculate fundamental frequency

- This chapter contains few examples of how cepstrum can be used to calculate fundamental frequency.
- Fundamental frequency exists only in signal which has harmonics.
- Harmonics are set of frequencies which are integer number of fundamental frequency.
- For instance, if fundamental frequency is 50 Hz it's harmonics would be 100, 150, 200, 250, ... Hz.
- It is possible that signal contains only few harmonics and that it doesn't contain their fundamental frequency. Such example would be signal containing frequences 150, 200 and 250 Hz but not the fundamental frequency of 50 Hz.

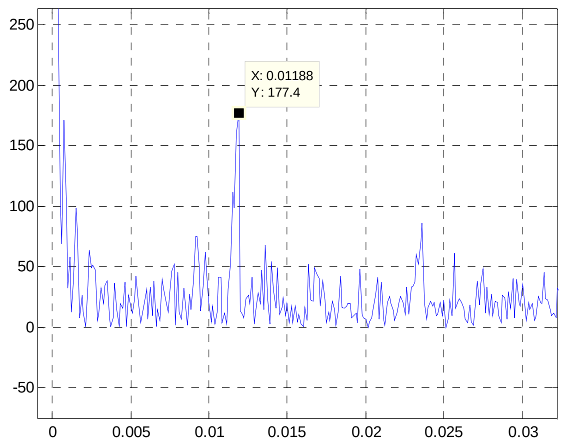
- Using MATLAB, cepstrum is basicly calculated like this:

```
%GET SOUND.
load GuitarSounds_8000;
x      = sound1(17400:18400-1);
Ts     = 1/8000;

%CEPSTRUM.
Y      = fft(x.*hamming(length(x)));
C      = fft(log(abs(Y)+eps));

%PLOT.
f_cep = Ts*(1:length(C));  %fdft=Ts
plot(f_cep,abs(C));
```

- Once we get the plot of the cepstrum, as shown on following figure:



we need to find highest peak after starting slope which in the above example is at 0.01188.

- Cepstrum looks at signal's harmonics as periodic signal.
- Cepstrum has found that frequency of such signal is 0.01188 Hz which means that period is 1/0.01188=84.175.
- This means that distance between harmonics is 84.175 Hz and this is by definition fundamental frequency.
- Using haaming window or log(DFT) is not necessary.
- Also above procedure uses MATALB's fft function which returns redundand results maling cepstrum look diferently probably because DFT periodicity is taken into account.

Hamming Window
log
Extra coefficients of first DFT

1.1. First example of signal with one set of harmonics

- In this chapter we will use cepstrum to find fundamental frequency of following signal:

$$x(t) = 0.5 + \sin(2\omega t) + \sin(4\omega t) + \sin(6\omega t)\ \ ,\ \ T=0.01\ \ ,\ \ T_s=(T/6)/2$$

- Fundamental frequency is 200Hz and harmonics are 400Hz and 600Hz.
- Signal and it's components together with sampled values at T_s intervals are shown on figure 1.1.a).
- DFT of such signal is periodic as shown on figure 1.1.b).

- Idea behind cepstrum is to look at such periodic DFT as if it is some descrete signal as shown with red line on figure 1.1.b).
- Taking DFT of such signal results in spectrum as shown on figure 1.1.c).
- DFT has detected frequency 0.005 Hz.
- This means that distance between harmonics is T=1/0.005=200 Hz which is fundamental frequency of our signal.

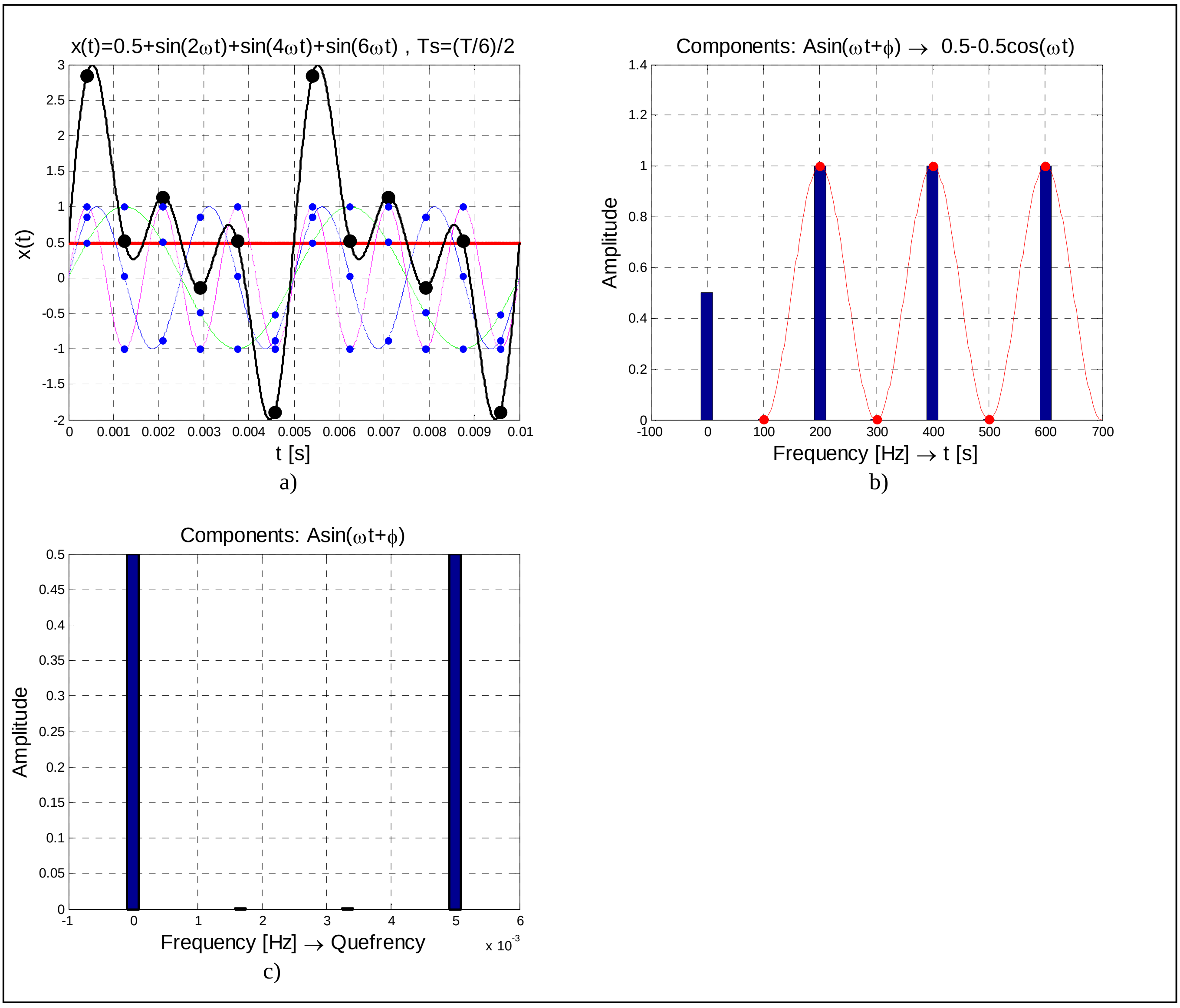


Figure 1.1. Using Cepstrum to calculate fundamental frequency of signal with one set of harmonics.

1.2. Second example of signal with one set of harmonics

- In this chapter we will use cepstrum to find fundamental frequency of following signal:

$$x(t) = 0.5 + \sin(3\omega t) + \sin(6\omega t) \ , \ T=0.01 \ , \ T_s=(T/6)/2$$

- Fundamental frequency is 300Hz and harmonic is 600Hz.
- Signal and it's components together with sampled values at T_s intervals are shown on figure 1.2.a).
- DFT of such signal is periodic as shown on figure 1.2.b).

- Idea behind cepstrum is to look at such periodic DFT as if it is some descrete signal as shown with red line on figure 1.2.b).
- Taking DFT of such signal results in spectrum as shown on figure 1.2.c).
- DFT has detected frequency 0.00333 Hz.
- This means that distance between harmonics is T=1/0.00333=300 Hz which is fundamental frequency of our signal.

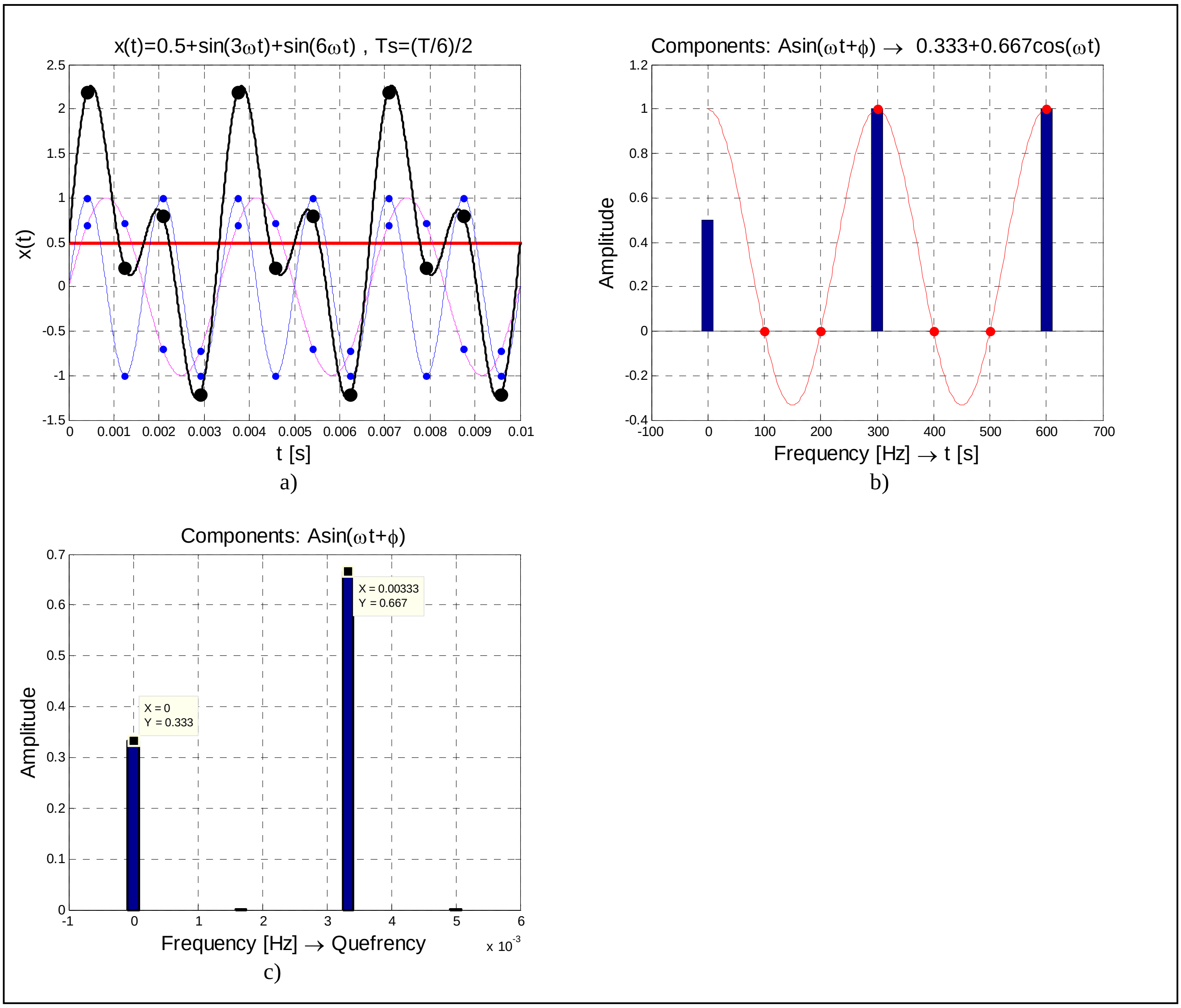


Figure 1.2. Using Cepstrum to calculate fundamental frequency in signal with one set of harmonics.

1.3. Signal with two sets of harmonics

– In this chapter we will use cepstrum to find fundamental frequencies of following signal:

$$x(t) = 0.5+[\sin(2\omega t)+\sin(4\omega t)+\sin(6\omega t)]+[\sin(3\omega t)+\sin(6\omega t)] \quad , \quad T=0.01 \quad , \quad T_s=(T/6)/2$$

– First fundamental frequency is 200Hz and harmonics are 400Hz and 600Hz.
– Second fundamental frequency is 300Hz and harmonic is 600Hz.
– Signal and it's components together with sampled values at T_s intervals are shown on figure 1.3.a).
– DFT of such signal is periodic as shown on figure 1.3.b).

– Idea behind cepstrum is to look at such periodic DFT as if it is two descrete signals as shown with red and blue lines on figure 1.3.b).
– Taking DFT of such signal results in spectrum as shown on figure 1.3.c).
– DFT has detected frequency 0.005Hz and 0.00333 Hz.
– This means that distance between:
  – first set of harmonics is T=1/0.005=200 Hz which is first fundamental frequency inside signal,
  – second set of harmonics is T=1/0.00333=300 Hz which is second fundamental frequency inside signal.

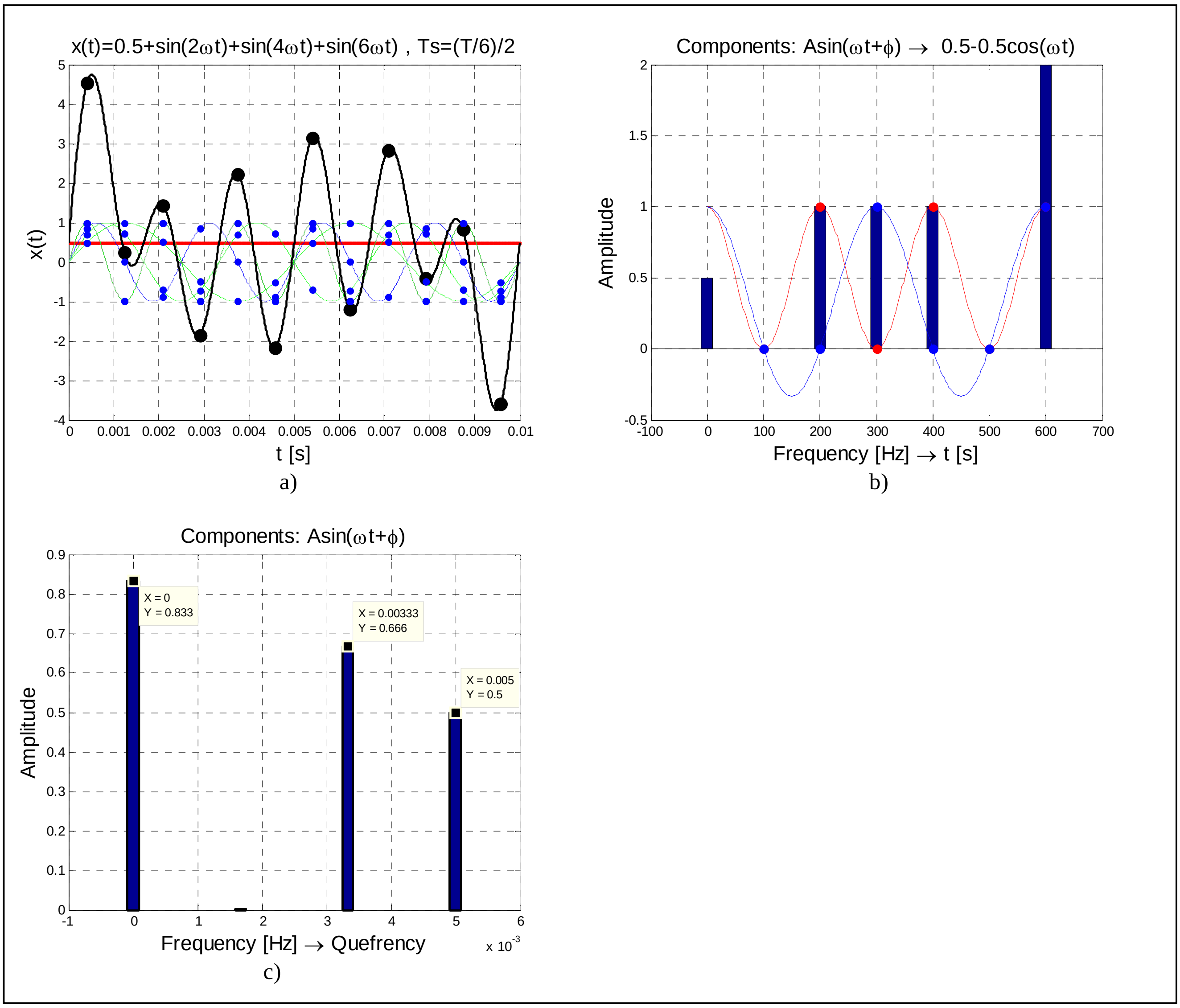
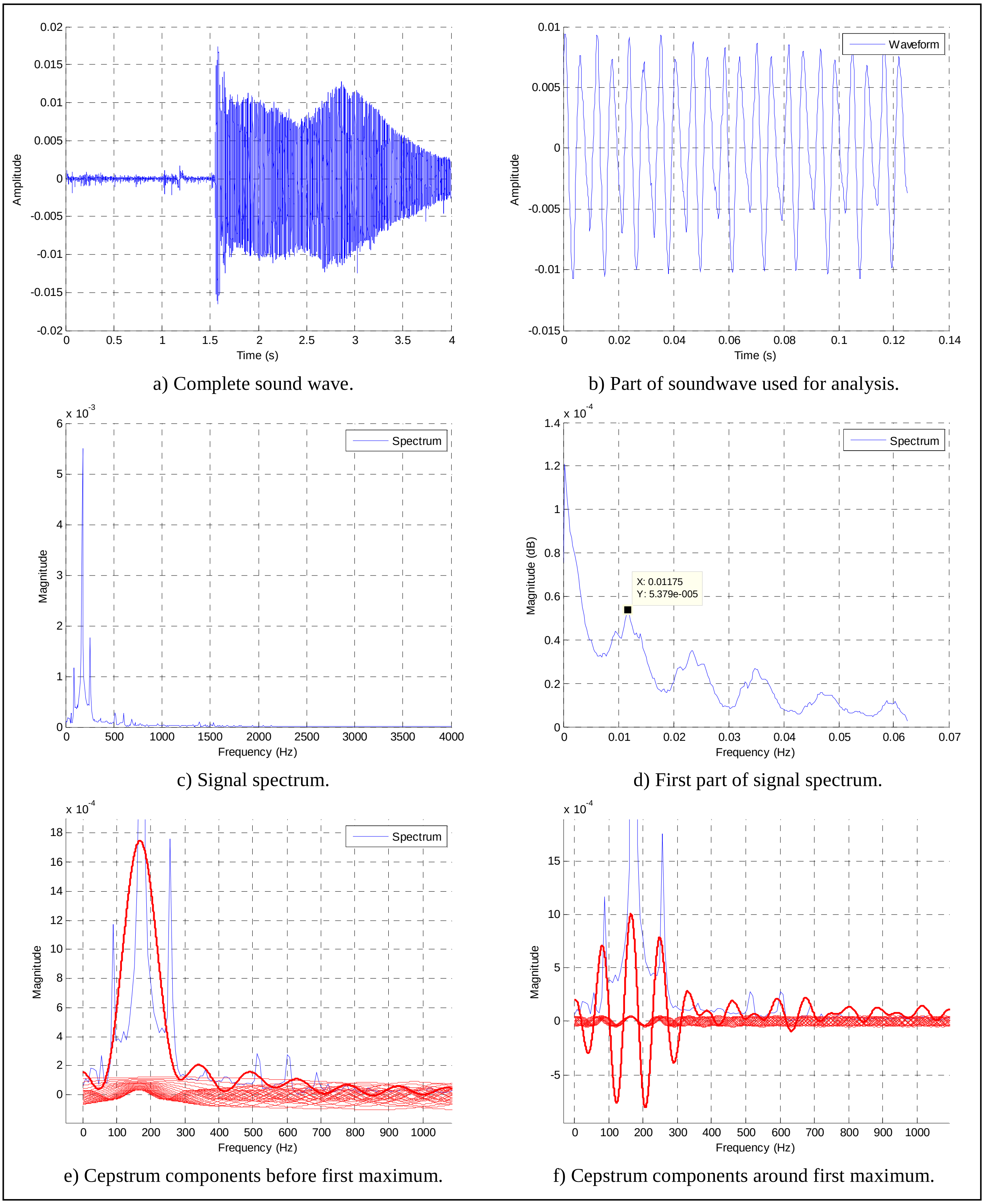


Figure 1.3. Using Cepstrum to calculate fundamental frequency in signal with harmonics.

1.4. Guitar string

- In this chapter we will find fundamental frequency of guitar string.
- Each guitar string must be properly tightened to produce predifened fundamental frequency before you can play guitar.
- Frequences of guitar strings in Hz should be: E=82.4, A=110, D=146.8, G=196, B=246.92, E=329.6

- Figure 1.4. shows how to find fundamental frequency of upper most guitar string which should have the frequency 82.4 Hz.



a) Complete sound wave.
b) Part of soundwave used for analysis.
c) Signal spectrum.
d) First part of signal spectrum.
e) Cepstrum components before first maximum.
f) Cepstrum components around first maximum.

Figure 1.4. Finding fundamental frequency of guitar string using cepstrum.

1.5. MATLAB functions

1.5.1. Figure 1.1.

```
%% MAIN SIGNAL PARAMETERS.-----------------------------------------------------------------------------------
clear;
T       = 0.01;         %Period of signal in seconds.
f       = 1/T;          %Frequency in periods per second.
w       = 2*pi/T;       %Frequency in radians per second.
Ts      = (T/6/2);      %Sample period in seconds is 2 samples per smallest period.
Ts_cont = (Ts)/100;     %To draw continuous signal we take 100 points between two samples.

%% CONTINUOUS SIGNAL.----------------------------------------------------------------------------------------

% DEFINE.
t_cont    = (0:Ts_cont:T);   %Time in seconds. Show 1 period of signal.
x_cont_0  = 0.5;
x_cont_1  = sin(2*w*t_cont);
x_cont_2  = sin(4*w*t_cont);
x_cont_3  = sin(6*w*t_cont);
x_cont    = x_cont_0+x_cont_1+x_cont_2+x_cont_3;

% DRAW.
figure(1);  hold off;
plot(t_cont, x_cont_0, 'r'); grid on; hold on;
plot(t_cont, x_cont_1, 'g');
plot(t_cont, x_cont_2, 'b');
plot(t_cont, x_cont_3, 'm');
plot(t_cont, x_cont,   'k','LineWidth',2);

% TITLE & LABELS.
h  = xlabel('t [s]');  set(h,'FontSize',15);
h  = ylabel('x(t)');   set(h,'FontSize',15);
h  = title('x(t)=0.5+sin(2\omegat)+sin(4\omegat)+sin(6\omegat) , Ts=(T/6)/2'); set(h,'FontSize',15);

%% DISCRETE SIGNAL.------------------------------------------------------------------------------------------

% TAKE SAMPLES FROM CONTINUOUS SIGNAL.
step = round(Ts/Ts_cont);
start= 50;  %To avoid that samples of fastest signal are always 0.
t    = t_cont  (start:step:length(t_cont)-1);
x_1  = x_cont_1(start:step:length(x_cont_1)-1);
x_2  = x_cont_2(start:step:length(x_cont_2)-1);
x_3  = x_cont_3(start:step:length(x_cont_3)-1);
x    = x_cont  (start:step:length(x_cont)-1);

% DRAW.
figure(1);
plot(t, x,   'k.','MarkerSize',30); grid on;   hold on;
plot(t, x_1, '.' ,'MarkerSize',15);
plot(t, x_2, '.' ,'MarkerSize',15);
plot(t, x_3, '.' ,'MarkerSize',15);
plot(t, x_3, '.' ,'MarkerSize',15);

%% DFT. -----------------------------------------------------------------------------------------------------
[a0,ak,bk]= DFT_Get_a0akbk_FromSamples(x);
[A0,Ak]    = DFT_Get_A0Ak_From_a0akbk(a0,ak,bk);
freq=(1:length(Ak))/Ts/length(x);
figure(2); hold off;
bar([0 freq],[A0 Ak],0.2);
h = xlabel(strcat('Frequency [Hz] \rightarrow t [s]')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); set(h,'FontSize',15); grid on; hold on;
h = title('Components: Asin(\omegat+\phi) \rightarrow  0.5-0.5cos(\omegat)'); set(h,'FontSize',15);

%% DEFINE CONTINUOUS SIGNAL.---------------------------------------------------------------------------------
T       = 200;            %Period of signal in seconds.
f       = 1/T;
w       = 2*pi/T;          %Frequency in radians per second.
Ts_cont = T/100;           %Sampling period is 100 points in smallest period.
t_cont  = (0:Ts_cont:3*T); %Time in seconds. Show 2 periods of signal.
x_cont  = 0.5-0.5*cos(w*t_cont);
figure(2);
plot(100+t_cont, x_cont,'r'); grid on; hold on;
x2   =Ak(1:length(Ak));          %So that we take full 3 periods of so-called signal.
freq2=freq(1:length(freq));
plot(freq2,x2,'r.','MarkerSize',20); grid on;

%% CEPSTRUM.------------------------------------------------------------------------------------------------
```

```
[ac0,ack,bck]= DFT_Get_a0akbk_FromSamples(x2);
[Ac0,Ack]     = DFT_Get_A0Ak_From_a0akbk(ac0,ack,bck);
freq=(1:length(Ack))/100/length(x2);
figure(4); bar([0 freq],[Ac0 Ack],0.1,'LineWidth',2);
h = xlabel(strcat('Frequency [Hz] \rightarrow Quefrency ')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); grid on; hold on;           set(h,'FontSize',15);
h = title('Components: Asin(\omegat+\phi)');                 set(h,'FontSize',15);
```

1.5.2. Figure 1.2.

```
%% MAIN SIGNAL PARAMETERS.---------------------------------------------------------------------------------
clear;
T       = 0.01;          %Period of signal in seconds.
f       = 1/T;           %Frequency in periods per second.
w       = 2*pi/T;        %Frequency in radians per second.
Ts      = (T/6/2);       %Sample period in seconds is 2 samples per smallest period.
Ts_cont = (Ts)/100;      %To draw continuous signal we take 100 points between two samples.

%% CONTINUOUS SIGNAL.--------------------------------------------------------------------------------------

% DEFINE.
t_cont    = (0:Ts_cont:T);   %Time in seconds. Show 1 period of signal.
x_cont_0  = 0.5;
y_cont_1  = sin(3*w*t_cont);
y_cont_2  = sin(6*w*t_cont);
x_cont    = x_cont_0+y_cont_1+y_cont_2;

% DRAW.
figure(1);  hold off;
plot(t_cont, x_cont_0, 'r'); grid on; hold on;
plot(t_cont, y_cont_1, 'm');
plot(t_cont, y_cont_2, 'b');
plot(t_cont, x_cont,   'k','LineWidth',2);

% TITLE & LABELS.
h  = xlabel('t [s]');  set(h,'FontSize',15);
h  = ylabel('x(t)');   set(h,'FontSize',15);
h  = title('x(t)=0.5+sin(3\omegat)+sin(6\omegat) , Ts=(T/6)/2'); set(h,'FontSize',15);

%% DISCRETE SIGNAL.----------------------------------------------------------------------------------------

% TAKE SAMPLES FROM CONTINUOUS SIGNAL.
step = round(Ts/Ts_cont);
start= 50;  %To avoid that samples of fastest signal are always 0.
t    = t_cont  (start:step:length(t_cont)-1);
y_1  = y_cont_1(start:step:length(y_cont_1)-1);
y_2  = y_cont_2(start:step:length(y_cont_2)-1);
x    = x_cont  (start:step:length(x_cont)-1);

% DRAW.
figure(1);
plot(t, x,   'k.','MarkerSize',30); grid on;   hold on;
plot(t, y_1, '.' ,'MarkerSize',15);
plot(t, y_2, '.' ,'MarkerSize',15);

%% DFT
[a0,ak,bk]= DFT_Get_a0akbk_FromSamples(x);
[A0,Ak]    = DFT_Get_A0Ak_From_a0akbk(a0,ak,bk);
freq=(1:length(Ak))/Ts/length(x);
figure(2); hold off;
bar([0 freq],[A0 Ak],0.2);
h = xlabel(strcat('Frequency [Hz] \rightarrow t [s]')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); set(h,'FontSize',15); grid on; hold on;
h = title('Components: Asin(\omegat+\phi) \rightarrow  0.333+0.667cos(\omegat)'); set(h,'FontSize',15);

%% DEFINE CONTINUOUS SIGNAL.-------------------------------------------------------------------------------
T       = 300;            %Period of signal in seconds.
f       = 1/T;
w       = 2*pi/T;          %Frequency in radians per second.
Ts_cont = T/100;           %Sampling period is 100 points in smallest period.
t_cont  = (0:Ts_cont:2*T); %Time in seconds. Show 2 periods of signal.
x_cont  = 0.333+0.666*cos(w*t_cont);
figure(2);
plot(t_cont, x_cont,'r'); grid on; hold on;
x2   =Ak(1:length(Ak));         %So that we take full 3 periods of so-called signal.
freq2=freq(1:length(freq));
plot(freq2,x2,'r.','MarkerSize',20); grid on;

%% CEPSTRUM.-----------------------------------------------------------------------------------------------
```

```
[ac0,ack,bck]= DFT_Get_a0akbk_FromSamples(x2);
[Ac0,Ack]    = DFT_Get_A0Ak_From_a0akbk(ac0,ack,bck);
freq=(1:length(Ack))/100/length(x2);
figure(4); bar([0 freq],[Ac0 Ack],0.1,'LineWidth',2);
h = xlabel(strcat('Frequency [Hz] \rightarrow Quefrency ')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); grid on; hold on;          set(h,'FontSize',15);
h = title('Components: Asin(\omegat+\phi)');                set(h,'FontSize',15);
```

1.5.3. Figure 1.3.

```
%% MAIN SIGNAL PARAMETERS.------------------------------------------------------------------------------
clear;
T       = 0.01;         %Period of signal in seconds.
f       = 1/T;          %Frequency in periods per second.
w       = 2*pi/T;       %Frequency in radians per second.
Ts      = (T/6/2);      %Sample period in seconds is 2 samples per smallest period.
Ts_cont = (Ts)/100;     %To draw continuous signal we take 100 points between two samples.

%% CONTINUOUS SIGNAL.-----------------------------------------------------------------------------------

% DEFINE.
t_cont    = (0:Ts_cont:T);   %Time in seconds. Show 1 period of signal.
x_cont_0  = 0.5;
x_cont_1  = sin(2*w*t_cont);  y_cont_1  = sin(3*w*t_cont);
x_cont_2  = sin(4*w*t_cont);
x_cont_3  = sin(6*w*t_cont);  y_cont_2  = sin(6*w*t_cont);
x_cont    = x_cont_0+x_cont_1+x_cont_2+x_cont_3+y_cont_1+y_cont_2;

% DRAW.
figure(1);  hold off;
plot(t_cont, x_cont_0, 'r'); grid on; hold on;
plot(t_cont, x_cont_1, 'g'); plot(t_cont, y_cont_1, 'g');
plot(t_cont, x_cont_2, 'b');
plot(t_cont, x_cont_3, 'm'); plot(t_cont, y_cont_2, 'g');
plot(t_cont, x_cont,   'k','LineWidth',2);

% TITLE & LABELS.
h  = xlabel('t [s]');  set(h,'FontSize',15);
h  = ylabel('x(t)');   set(h,'FontSize',15);
h  = title('x(t)=0.5+sin(2\omegat)+sin(4\omegat)+sin(6\omegat) , Ts=(T/6)/2'); set(h,'FontSize',15);

%% DISCRETE SIGNAL.-------------------------------------------------------------------------------------

% TAKE SAMPLES FROM CONTINUOUS SIGNAL.
step = round(Ts/Ts_cont);
start= 50;  %To avoid that samples of fastest signal are always 0.
t    = t_cont  (start:step:length(t_cont)-1);
x_1  = x_cont_1(start:step:length(x_cont_1)-1);   y_1  = y_cont_1(start:step:length(y_cont_1)-1);
x_2  = x_cont_2(start:step:length(x_cont_2)-1);
x_3  = x_cont_3(start:step:length(x_cont_3)-1);   y_2  = y_cont_2(start:step:length(y_cont_2)-1);
x    = x_cont  (start:step:length(x_cont)-1);

% DRAW.
figure(1);
plot(t, x,   'k.','MarkerSize',30); grid on;   hold on;
plot(t, x_1, '.' ,'MarkerSize',15); plot(t, y_1, '.' ,'MarkerSize',15);
plot(t, x_2, '.' ,'MarkerSize',15);
plot(t, x_3, '.' ,'MarkerSize',15); plot(t, y_2, '.' ,'MarkerSize',15);
plot(t, x_3, '.' ,'MarkerSize',15);

%% DFT
[a0,ak,bk]= DFT_Get_a0akbk_FromSamples(x);
[A0,Ak]   = DFT_Get_A0Ak_From_a0akbk(a0,ak,bk);
freq=(1:length(Ak))/Ts/length(x);
figure(2); hold off;
bar([0 freq],[A0 Ak],0.2);
h = xlabel(strcat('Frequency [Hz] \rightarrow t [s]')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); set(h,'FontSize',15); grid on; hold on;
h = title('Components: Asin(\omegat+\phi) \rightarrow  0.5-0.5cos(\omegat)'); set(h,'FontSize',15);
x_dft =Ak(1:length(Ak));         %So that we take full 3 periods of so-called signal.

%% DEFINE 200Hz CONTINUOUS SIGNAL.----------------------------------------------------------------------
------

% SIGNAL PARAMETERS.
Tx       = 200;            %Period of signal in seconds.
fx       = 1/Tx;
wx       = 2*pi/Tx;          %Frequency in radians per second.
Tsx_cont = Tx/100;           %Sampling period is 100 points in smallest period.
```

```
tx_cont  = (0:Tsx_cont:3*Tx); %Time in seconds. Show 2 periods of signal.
x_cont   = 0.5+0.5*cos(wx*tx_cont);

% DRAW CONTINUOUS SIGNAL.
figure(2);
plot(tx_cont, x_cont,'r'); grid on; hold on;

% DRAW DISCRETE SAMPLES.
x2   = [0 1 0 1 0 1]';         %So that we take full 3 periods of so-called signal.
freq2=freq(1:length(freq));
plot(freq2,x2,'r.','MarkerSize',20); grid on;

%% DEFINE 300Hz CONTINUOUS SIGNAL.-------------------------------------------------------------------------
------

% SIGNAL PARAMETERS.
Ty       = 300;            %Period of signal in seconds.
fy       = 1/Ty;
wy       = 2*pi/Ty;          %Frequency in radians per second.
Tsy_cont = Ty/100;           %Sampling period is 100 points in smallest period.
ty_cont  = (0:Tsy_cont:2*Ty); %Time in seconds. Show 2 periods of signal.
y_cont  = 0.333+0.666*cos(wy*ty_cont);

% DRAW CONTINUOUS SIGNAL.
figure(2);
plot(ty_cont, y_cont,'b'); grid on; hold on;

% DRAW DISCRETE SAMPLES.
y2   = [0 0 1 0 0 1]';         %So that we take full 3 periods of so-called signal.
freq2=freq(1:length(freq));
plot(freq2,y2,'b.','MarkerSize',20); grid on;


%% CEPSTRUM.-----------------------------------------------------------------------------------------------
[ac0,ack,bck]= DFT_Get_a0akbk_FromSamples(x_dft);
[Ac0,Ack]    = DFT_Get_A0Ak_From_a0akbk(ac0,ack,bck);
freq=(1:length(Ack))/100/length(x2);
figure(4); bar([0 freq],[Ac0 Ack],0.1,'LineWidth',2);
h = xlabel(strcat('Frequency [Hz] \rightarrow Quefrency ')); set(h,'FontSize',15);
h = ylabel(strcat('Amplitude')); grid on; hold on;           set(h,'FontSize',15);
h = title('Components: Asin(\omegat+\phi)');                 set(h,'FontSize',15);
```

1.5.4. Figure 1.4.

For recording sound you can also use:

```
%% RECORD SOUND.==========================================================

r = audiorecorder(8000,16,1); %Recorder uses sampling frequency 44100Hz and 8 bit mono encoding.
recordblocking(r,3);          %Use recorder to start recording sound for 3.5 seconds.
s = getaudiodata(r);          %Load recorded sound into array 's' using amplitude values [-1,1].

%% SOUND.=================================================================

%GET SECTION OF SOUND.----------------------------------------------------
x  = s(length(s)-1000:length(s)-1); %Take last 1000 samples.
fs = 8000;
Ts = 1/fs;
```

– **Figures were created using:**

```
%% CLEAR.=================================================================
clear;

%% SOUND.=================================================================

%GET SECTION OF SOUND.----------------------------------------------------
load GuitarSounds_8000;
x  = sound1(17400:18400-1);
fs = 8000;
Ts = 1/fs;

%PLOT COMPLETE SOUND.-----------------------------------------------------
t=(0:length(sound1)-1)*Ts;        % times of sampling instants
figure(1); clf(1); grid on; hold on;
plot(t,sound1); xlabel('Time (s)'); ylabel('Amplitude');

%PLOT PARTIAL SOUND.------------------------------------------------------
t=(0:length(x)-1)*Ts;        % times of sampling instants
figure(2); clf(2); grid on; hold on;
plot(t,x); legend('Waveform'); xlabel('Time (s)'); ylabel('Amplitude');

%% DFT.===================================================================

%CALCULATE DFT OF WINDOWED SIGNAL.----------------------------------------
%x=x.*hamming(length(x));
[C0,Ck] = i_DFT(x);

%PREPARE TO PLOT SPECTRUM.------------------------------------------------
Tdft = Ts*length(x);
fdft = 1/Tdft;
f    = fdft*(1:length(Ck));

%PLOT SPECTRUM AMPLITUDES.------------------------------------------------
figure(3); clf(3); grid on; hold on;
plot([0 f],[C0 abs(Ck)]);
legend('Spectrum'); xlabel('Frequency (Hz)'); ylabel('Magnitude');

%% CEPSTRUM..=============================================================

%CEPSTRUM.----------------------------------------------------------------
[C0_cep,Ck_cep] = i_DFT(abs(Ck));

%PREPARE TO PLOT SPECTRUM.------------------------------------------------
Tdft_cep = fdft*length(Ck);
fdft_cep = 1/Tdft_cep;
f_cep    = fdft_cep*(1:length(Ck_cep));

%PLOT SPECTRUM AMPLITUDES.------------------------------------------------
figure(4); clf(4); grid on; hold on;
plot([0 f_cep],[C0_cep abs(Ck_cep)]);
legend('Spectrum'); xlabel('Frequency (Hz)'); ylabel('Magnitude (dB)');

%% FIND FUNDAMENTAL FREQUENCY.============================================

%FIND FIRST MINIMUM.------------------------------------------------------
temp=abs(Ck_cep(1));
for k=2:length(Ck_cep)
  if abs(Ck_cep(k))<temp
    temp = abs(Ck_cep(k));
  else break;
```

```
  end
end

%FIND MAXIMUM IN THE REST OF CEPSTRUM.-------------------------------------
[cep,maxk]=max(abs(Ck_cep(k:length(Ck_cep))));
1/f_cep(k+maxk-1)
```