DBBT
Digital Broadcasting & Broadband Technologies

Co-funded by the Erasmus+ Programme of the European Union

Procesiranje signala

Profesor dr Miroslav Lutovac

"This project has been funded with support from the European Commission. This publication [communication] reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein"

Sadržaj predmeta, Teorijska nastava

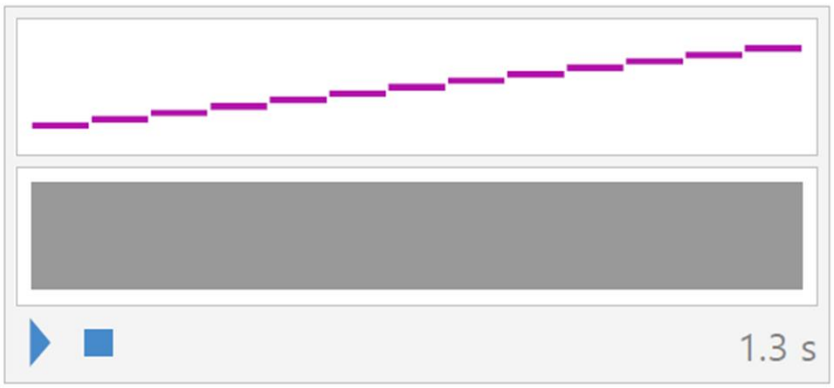
1. Uvodno predavanje. Upoznavanje sa planom i programom, ciljevima, ishodom i metodama.
2. Šta je procesiranje signala, istorijski pregled obrade signala, primeri primene.
3. Vizuelizacija signala (Python, Excel).
4. Kompleksni ekponencijalni diskretni signali. Primer sinteze muzičkog signala.
5. Furijeova analiza: Diskretna Furijeova transformacija (DFT) i serija (DFS). Brza Furijeova transformacija (Fast Fourier transform, FFT) i primena za spektralne analizatore i osciloskope.
6. Linearani filtri: konvolucija, idealni i realni filtri, dizajn filtra. Primena konvolucije u GPS sistemima.

…

Sinteza muzičkog signala

- SoundNote[0] represents a music-like sound note with the specified pitch.
- 0, middle C
- SoundNote[n] , n semitones from middle C
- "C", "C#", "D", ... notes in the middle C octave

```
sound01 = Sound[Table[SoundNote[i, 0.1, "Violin"], {i, 0, 12}]]
EmitSound[sound01]
Export["sound01.WAV", sound01]
```

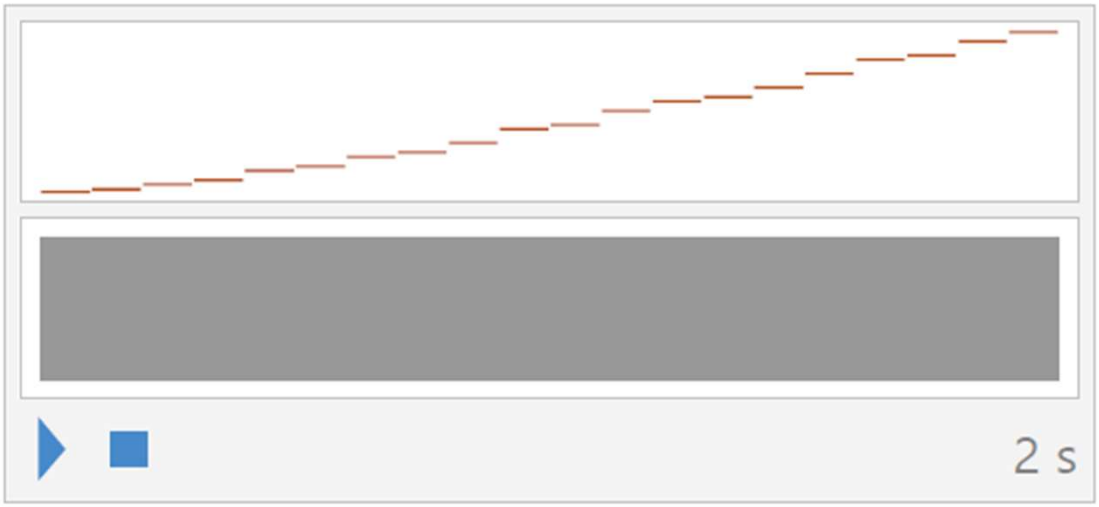


sound01.WAV

Sinteza muzičkog signala

- SoundNote[0,t] takes the note to have duration t
- SoundNote[0,t,"Violin"] takes the note to be in the specified style

```
sound02 = Sound[Table[SoundNote[Prime[i] - 20, 0.1, "Woodblock"], {i, 20}]]
EmitSound[sound02]
Export["sound02.WAV", sound02]
```

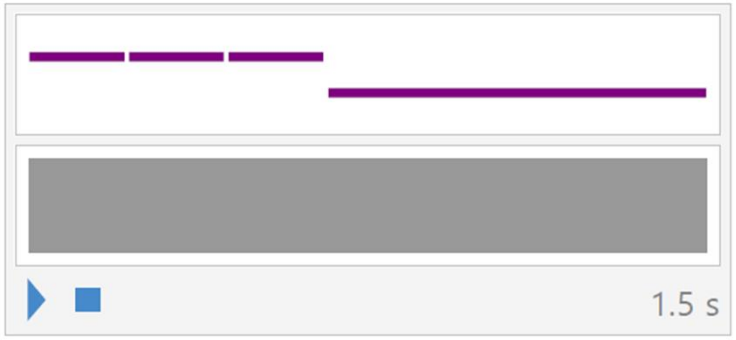


sound02.WAV

Sinteza muzičkog signala

- SoundNote[0] represents a music-like sound note with the specified pitch.

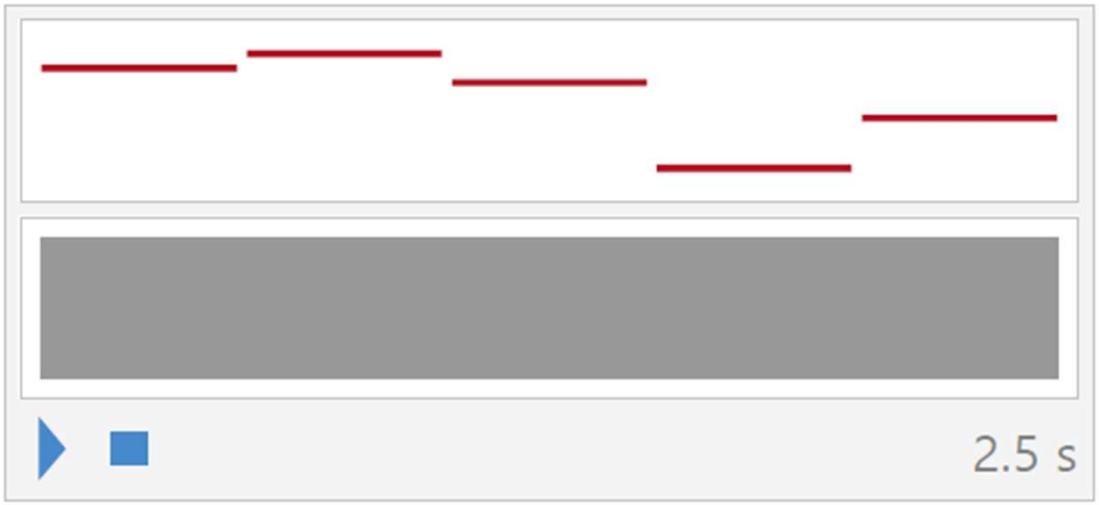
```
sound03 = Sound[{SoundNote["G"], SoundNote["G"], SoundNote["G"], SoundNote["Eb", 4]}, 1.5]
EmitSound[sound03]
Export["sound03.WAV", sound03]
```



sound03.WAV

Sinteza muzičkog signala

```
sound04 = Sound[SoundNote[#, 0.5, "Polysynth"] & /@ {2, 4, 0, -12, -5}]
EmitSound[sound04]
Export["sound04.WAV", sound04]
```



sound04.WAV

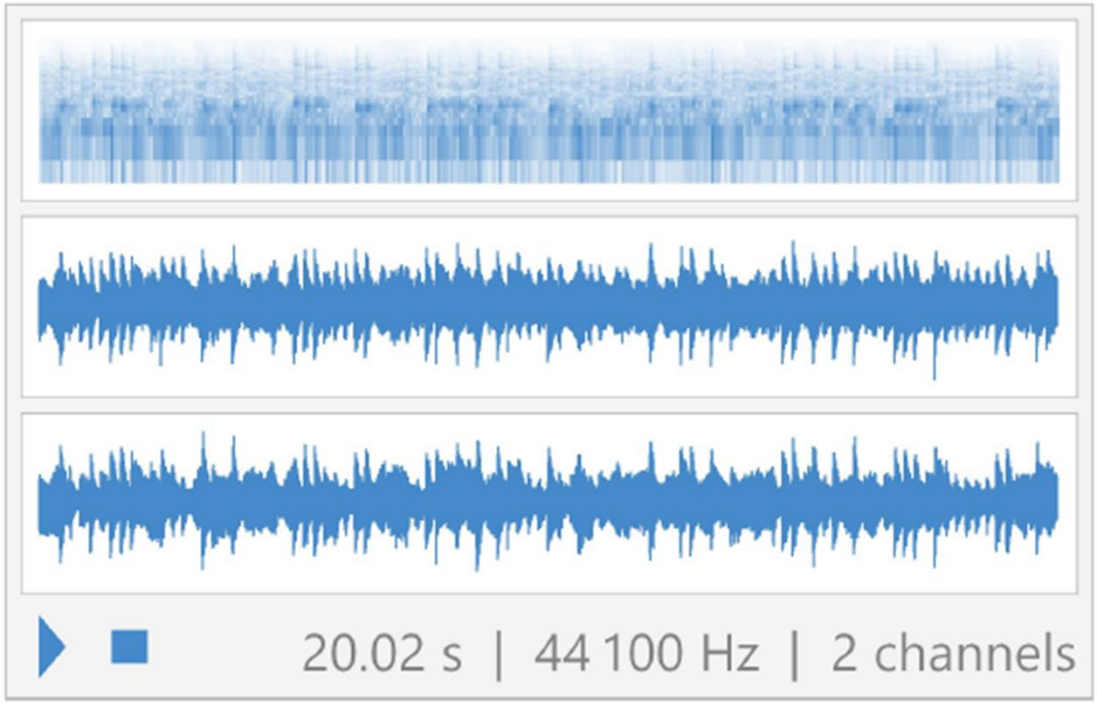
Sinteza muzičkog signala

```
sound05 = Sound[Table[
    SoundNote[RandomInteger[7], .2, RandomChoice[{"Piano", "Guitar", "Violin"}]], 100]]
EmitSound[sound05]
Export["sound05.WAV", sound05]
```

sound05.WAV

Sinteza muzičkog signala

```
snd05 = Import["sound05.WAV", "Sound"]
```



Styles

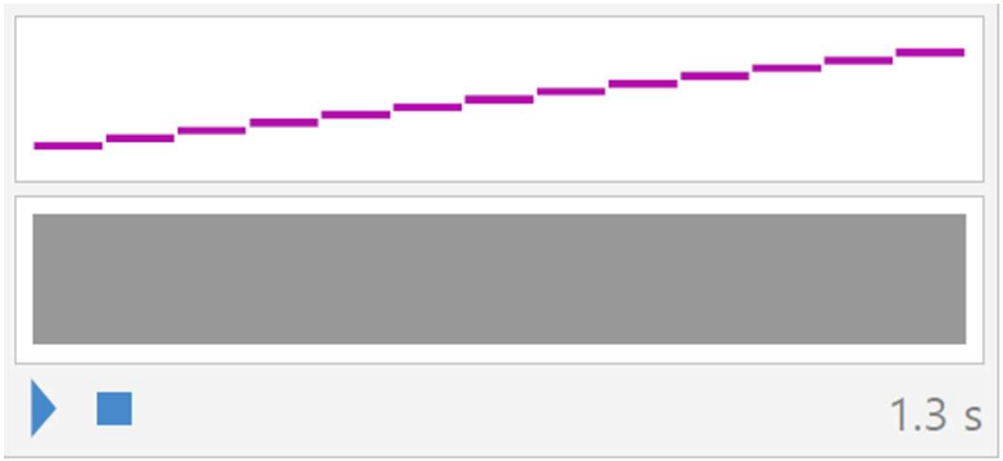
"Accordion"	"Agogo"	"AltoSax"	"Applause"
"Atmosphere"	"Bagpipe"	"Bandoneon"	"Banjo"
"BaritoneSax"	"Bass"	"BassAndLead"	"Bassoon"
"Bird"	"BlownBottle"	"Bowed"	"BrassSection"
"Breath"	"Brightness"	"BrightPiano"	"Calliope"
"Celesta"	"Cello"	"Charang"	"Chiff"
"Choir"	"Clarinet"	"Clavi"	"Contrabass"
"Crystal"	"DrawbarOrgan"	"Dulcimer"	"Echoes"
"ElectricBass"	"ElectricGrandPiano"	"ElectricGuitar"	"ElectricPiano"
"ElectricPiano2"	"EnglishHorn"	"Fiddle"	"Fifths"
"Flute"	"FrenchHorn"	"FretlessBass"	"FretNoise"

```
"Glockenspiel"      "Goblins"           "Guitar"            "GuitarDistorted"
"GuitarHarmonics"   "GuitarMuted"       "GuitarOverdriven"  "Gunshot"
"Halo"              "Harmonica"         "Harp"              "Harpsichord"
"Helicopter"        "HonkyTonkPiano"    "JazzGuitar"        "Kalimba"
"Koto"              "Marimba"           "MelodicTom"        "Metallic"
"MusicBox"          "MutedTrumpet"      "NewAge"            "Oboe"
"Ocarina"           "OrchestraHit"      "Organ"             "PanFlute"
"PercussiveOrgan"   "Piano"             "Piccolo"           "PickedBass"
"PizzicatoStrings"  "Polysynth"         "Rain"              "Recorder"
"ReedOrgan"         "ReverseCymbal"     "RockOrgan"         "Sawtooth"
"SciFi"             "Seashore"          "Shakuhachi"        "Shamisen"
"Shanai"            "Sitar"             "SlapBass"          "SlapBass2"
"SopranoSax"        "Soundtrack"        "Square"            "Steeldrums"
"SteelGuitar"       "Strings"           "Strings2"          "Sweep"
"SynthBass"         "SynthBass2"        "SynthBrass"        "SynthBrass2"
"SynthDrum"         "SynthStrings"      "SynthStrings2"     "SynthVoice"
"Taiko"             "Telephone"         "TenorSax"          "Timpani"
"Tinklebell"        "TremoloStrings"    "Trombone"          "Trumpet"
"Tuba"              "TubularBells"      "Vibraphone"        "Viola"
"Violin"            "Voice"             "VoiceAahs"         "VoiceOohs"
"Warm"              "Whistle"           "Woodblock"         "Xylophone"
```

```
"BassDrum"       "BassDrum2"      "BellTree"       "Cabasa"
"Castanets"      "ChineseCymbal"  "Clap"           "Claves"
"Cowbell"        "CrashCymbal"    "CrashCymbal2"   "ElectricSnare"
"GuiroLong"      "GuiroShort"     "HighAgogo"      "HighBongo"
"HighCongaMute"  "HighCongaOpen"  "HighFloorTom"   "HighTimbale"
"HighTom"        "HighWoodblock"  "HiHatClosed"    "HiHatOpen"
"HiHatPedal"     "JingleBell"     "LowAgogo"       "LowBongo"
"LowConga"       "LowFloorTom"    "LowTimbale"     "LowTom"
"LowWoodblock"   "Maracas"        "MetronomeBell"  "MetronomeClick"
"MidTom"         "MidTom2"        "MuteCuica"      "MuteSurdo"
"MuteTriangle"   "OpenCuica"      "OpenSurdo"      "OpenTriangle"
"RideBell"       "RideCymbal"     "RideCymbal2"    "ScratchPull"
"ScratchPush"    "Shaker"         "SideStick"      "Slap"
"Snare"          "SplashCymbal"   "SquareClick"    "Sticks"
"Tambourine"     "Vibraslap"      "WhistleLong"    "WhistleShort"
```

sound0x

```
clear all, close all, clc
[y,fs]=audioread ("wav/sound05.wav");
y1=y(:,1);y2=y(:,2);
sound(y1,fs); pause(.2);sound(y2,fs)
t=1:length(y1);
t=t/fs;
stem(t,y1,'.')
xlabel('vreme (s)');
ylabel('normalizovan signal');
title(['sound05, Fs = ' num2str(fs)]);figure
stem(t,y2,'.')
xlabel('vreme (s)');
ylabel('normalizovan signal');
title(['sound05, Fs = ' num2str(fs)]);
```


1.3 s

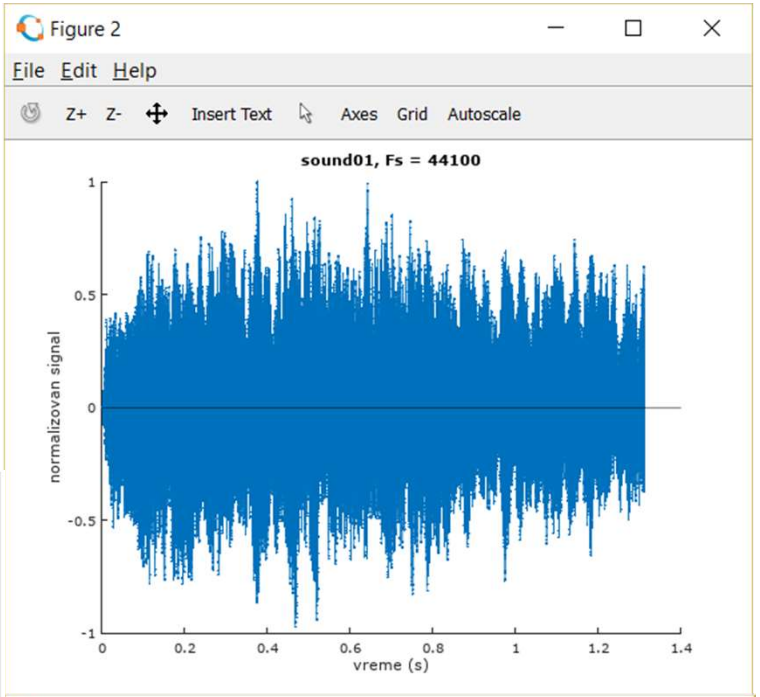


Figure 2
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound01, Fs = 44100
normalizovan signal
vreme (s)

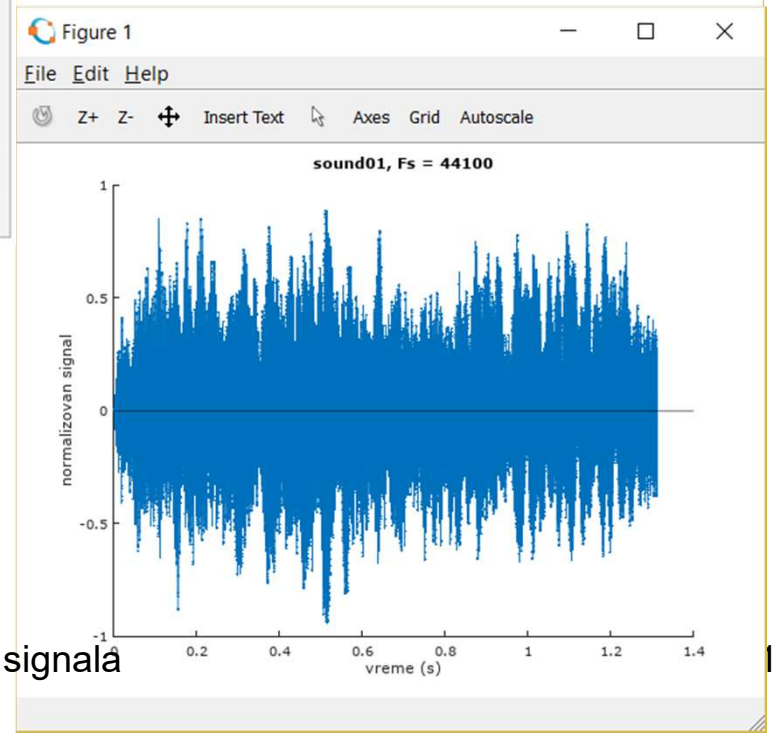
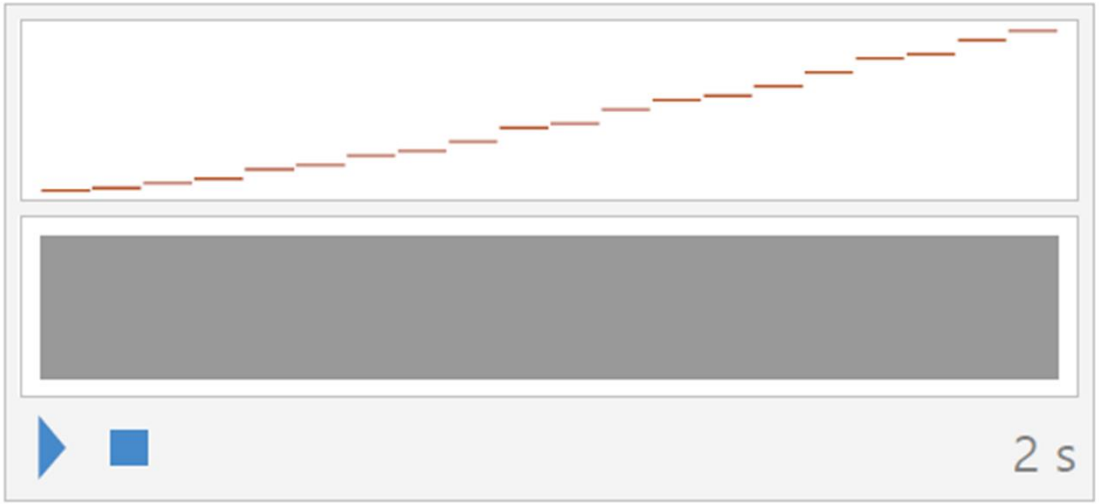


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound01, Fs = 44100
normalizovan signal
vreme (s)


2 s

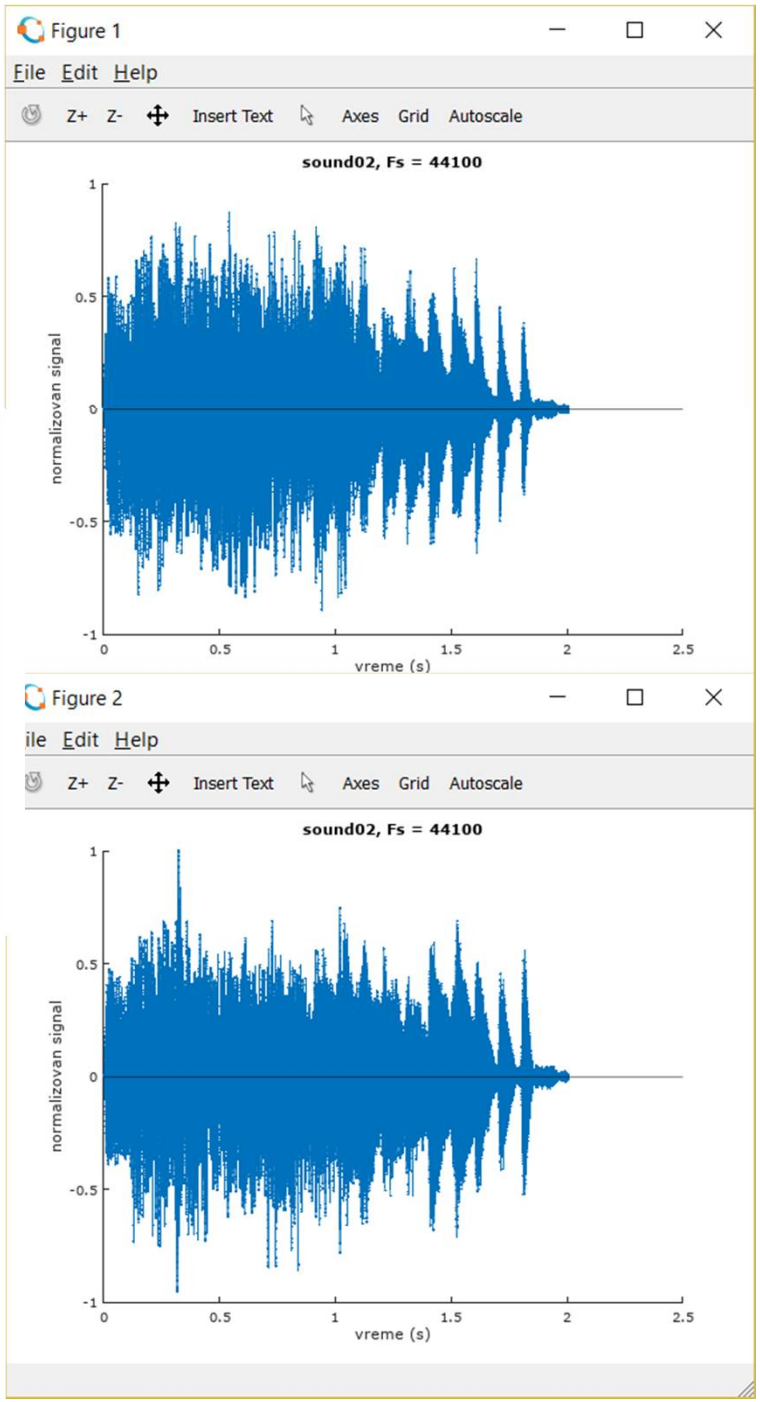
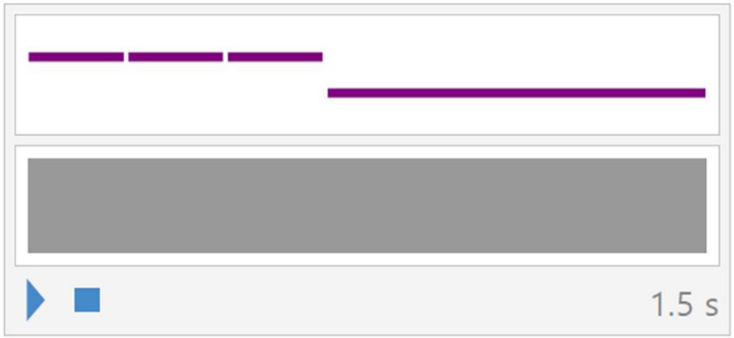
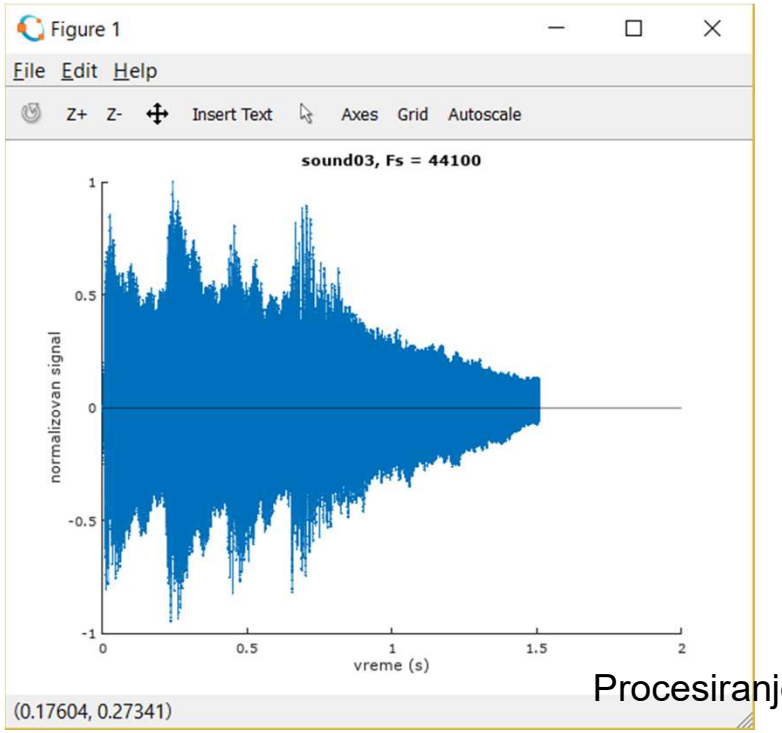


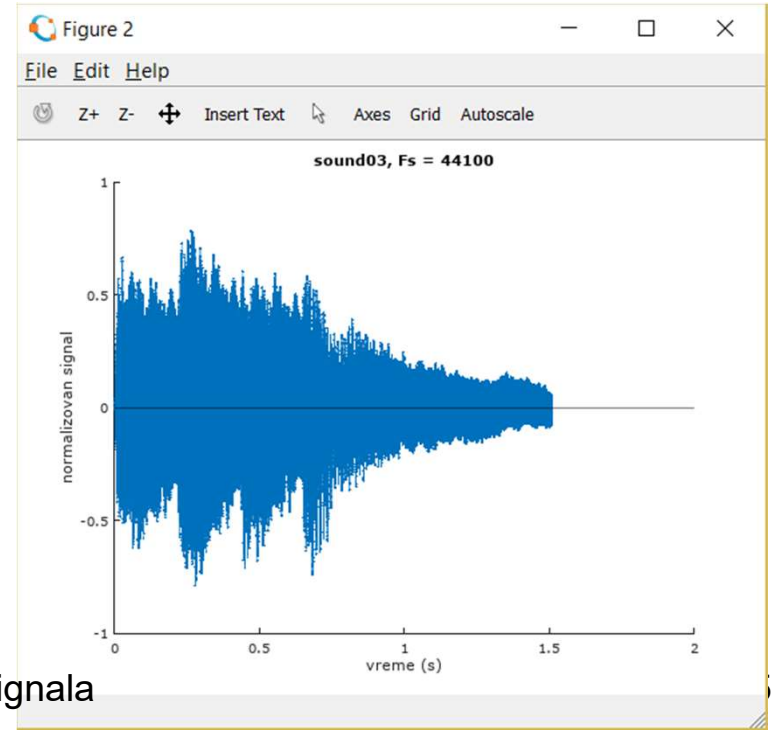
Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound02, Fs = 44100
normalizovan signal
vreme (s)
Figure 2
ile Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound02, Fs = 44100
normalizovan signal
vreme (s)

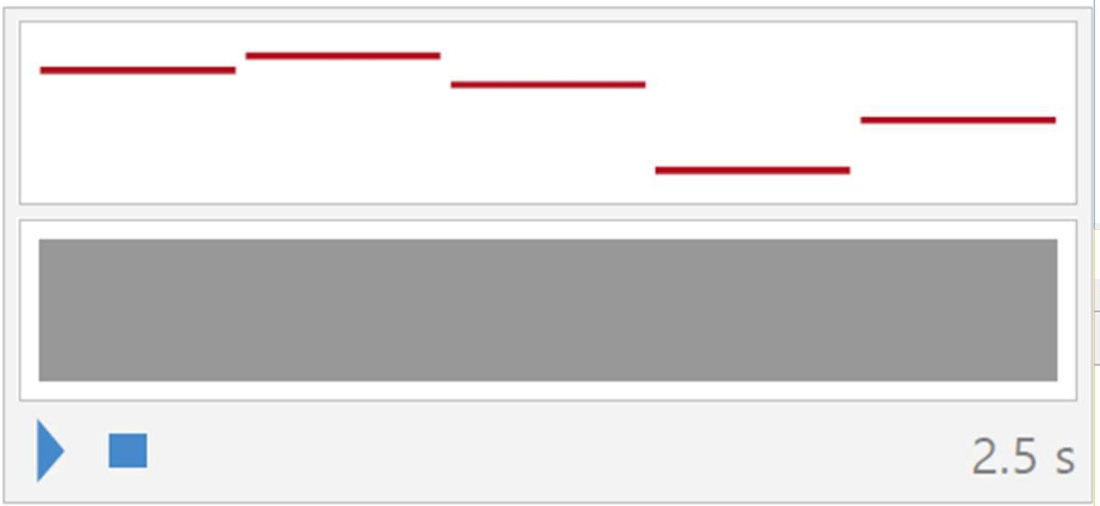
```
sound03 = Sound[{SoundNote["G"], SoundNote["G"], SoundNote["G"], SoundNote["Eb", 4]}, 1.5]
EmitSound[sound03]
Export["sound03.WAV", sound03]
```



sound03.WAV






2.5 s

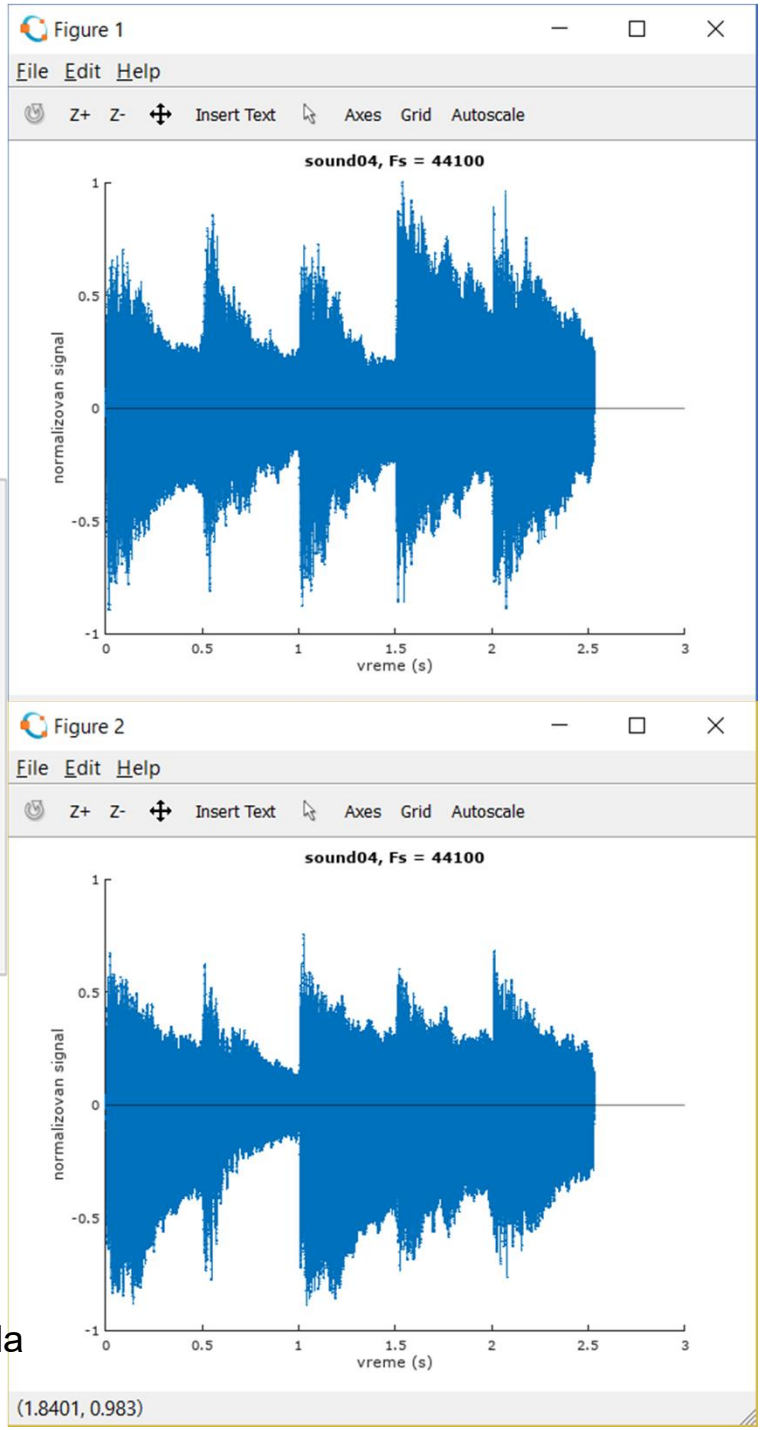
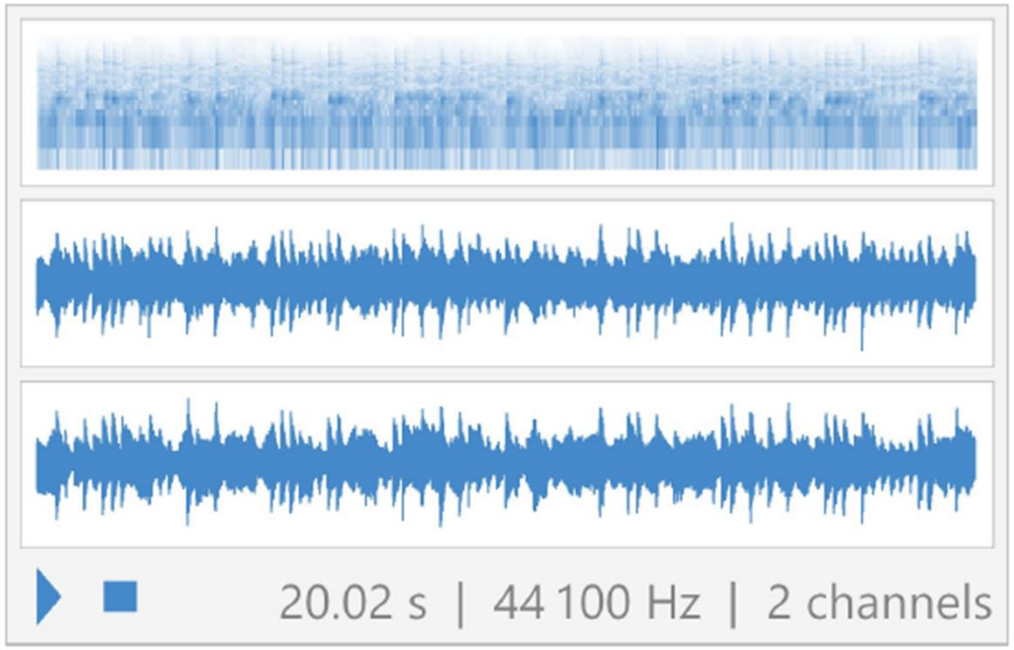
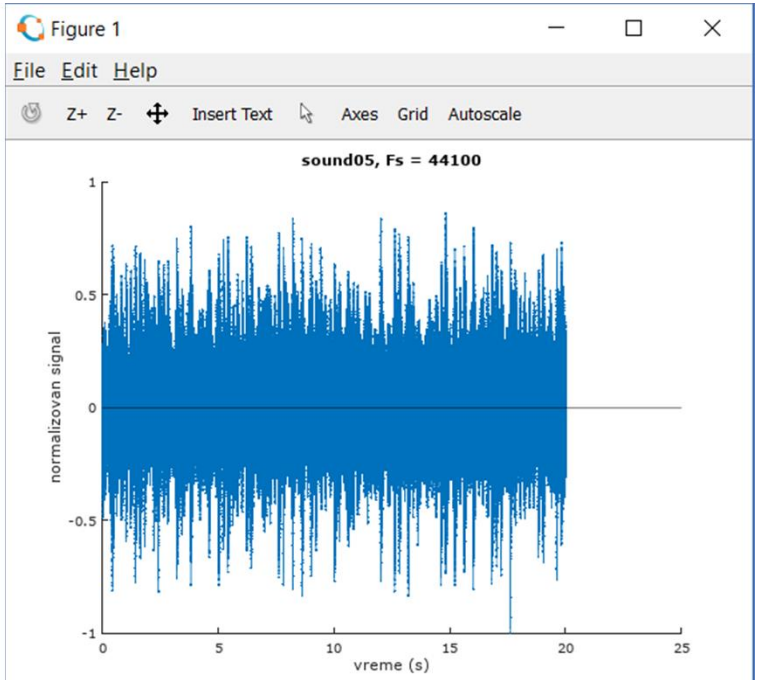


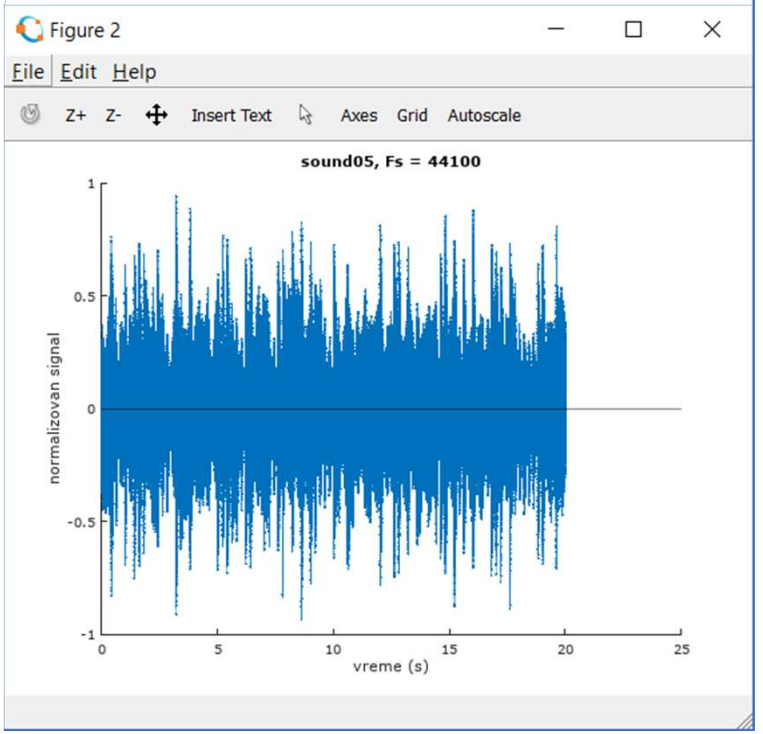
Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound04, Fs = 44100
normalizovan signal
vreme (s)
Figure 2
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound04, Fs = 44100
normalizovan signal
vreme (s)
(1.8401, 0.983)

sound05.WAV

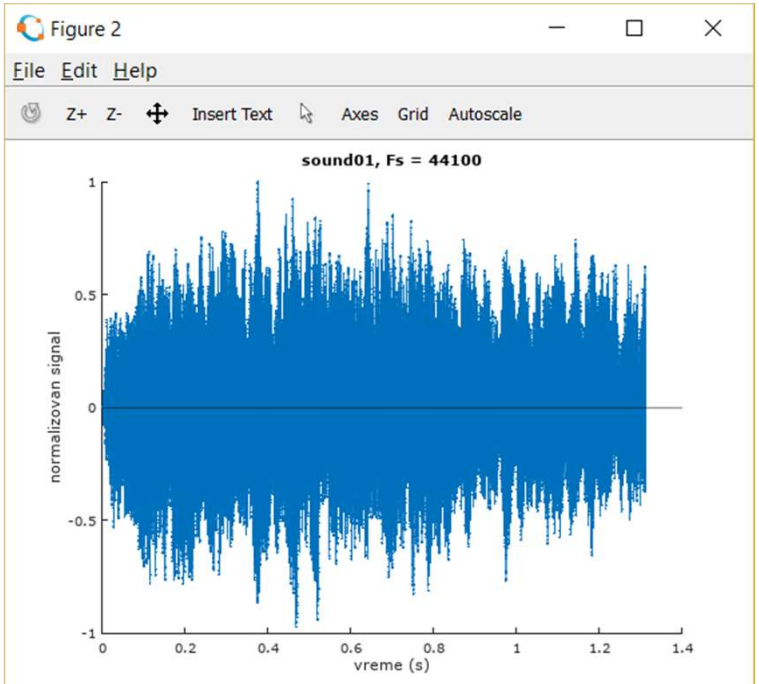
```
snd05 = Import["sound05.WAV", "Sound"]
```

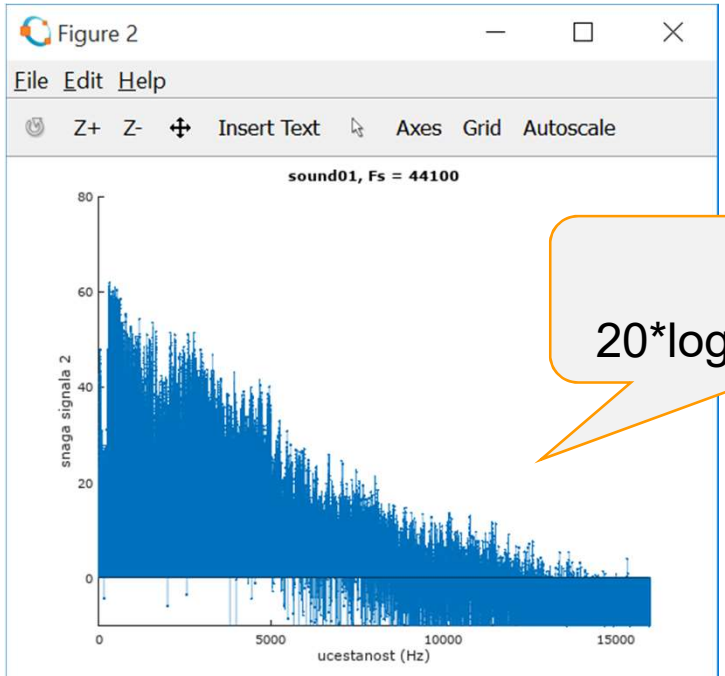






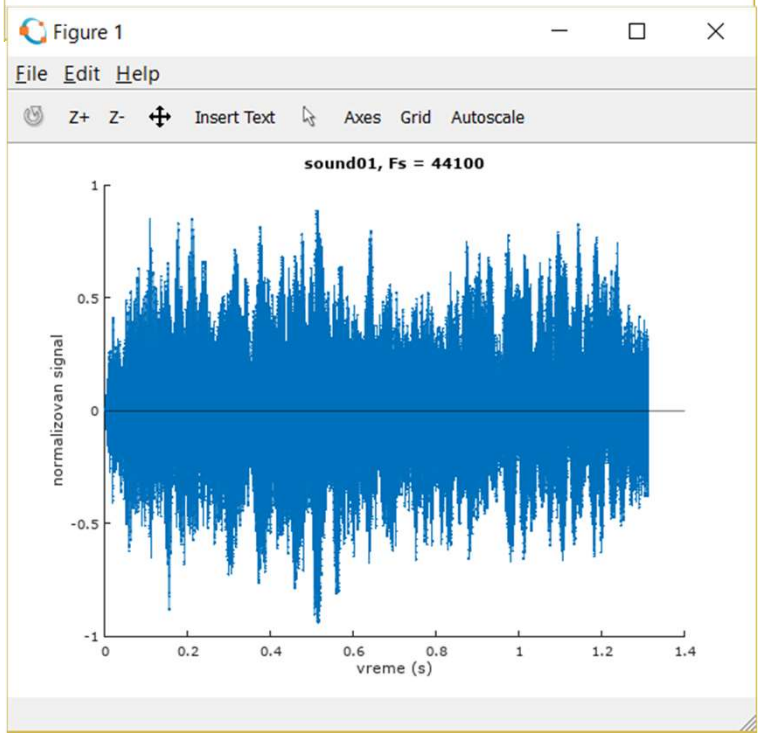
```
t=1:length(y1);
t=t/fs;
x1=fft(y1);
N1= length(y1)-1;
n= 0:N1;
a1 = 20*log10(abs(x1));
stem(fs*n/N1,a1,'.')
xlabel('ucestanost (Hz)');
ylabel('snaga signala 1');
title(['sound01, Fs = ' num2str(fs)]);
ax = axis; ax(2)=16000; ax(3)=ax(4)-90; axis(ax)
figure
x2=fft(y2);
N2= length(y2)-1;
n= 0:N2;
a2 = 20*log10(abs(x2));
stem(fs*n/N1,a2,'.')
xlabel('ucestanost (Hz)');
ylabel('snaga signala 2');
title(['sound01, Fs = ' num2str(fs)]);
ax = axis; ax(2)=16000; ax(3)=ax(4)-90; axis(ax)
```

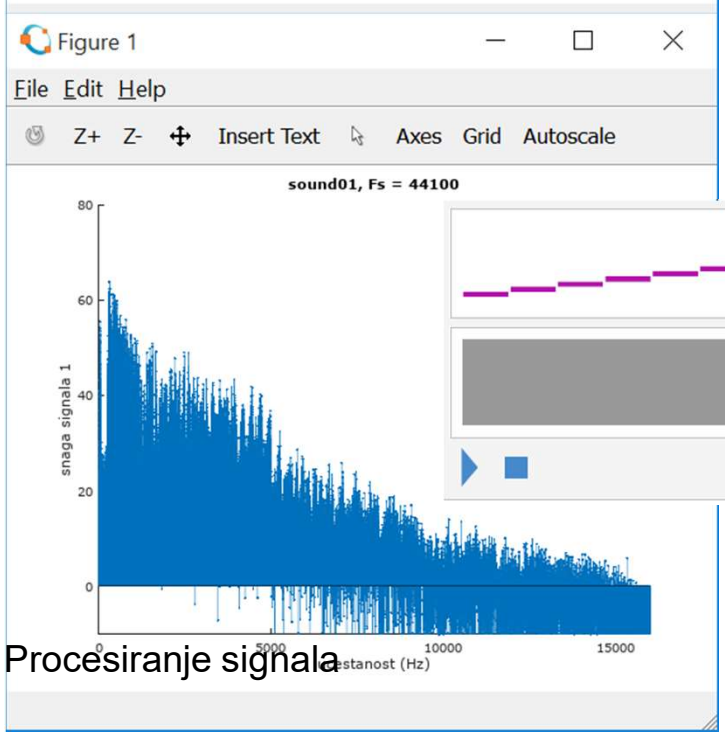


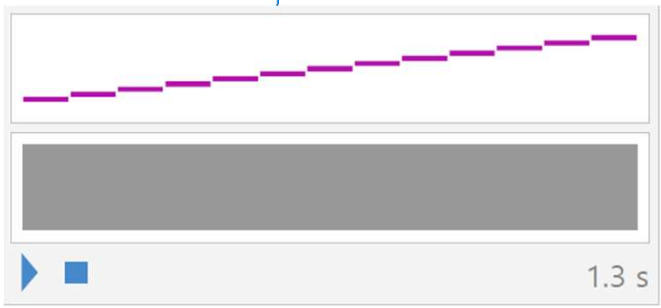


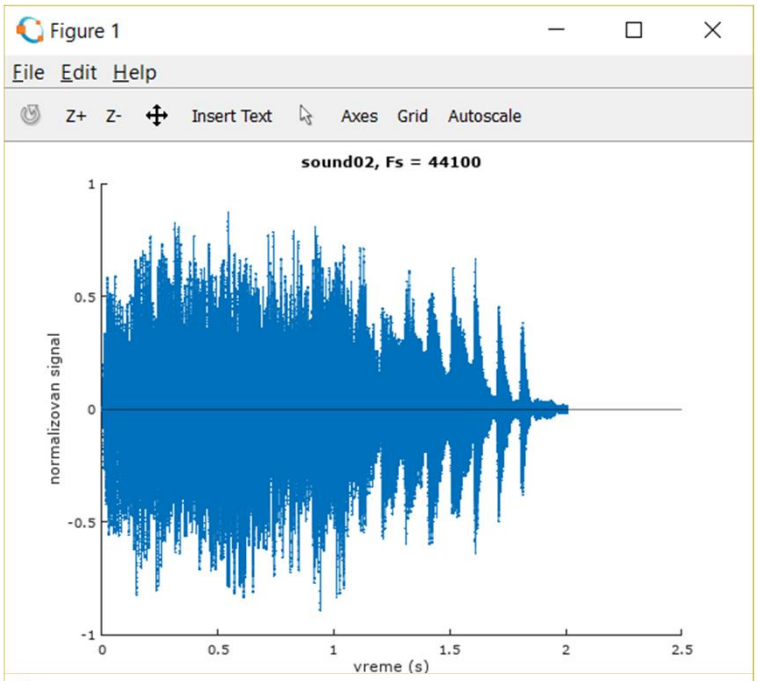
fft

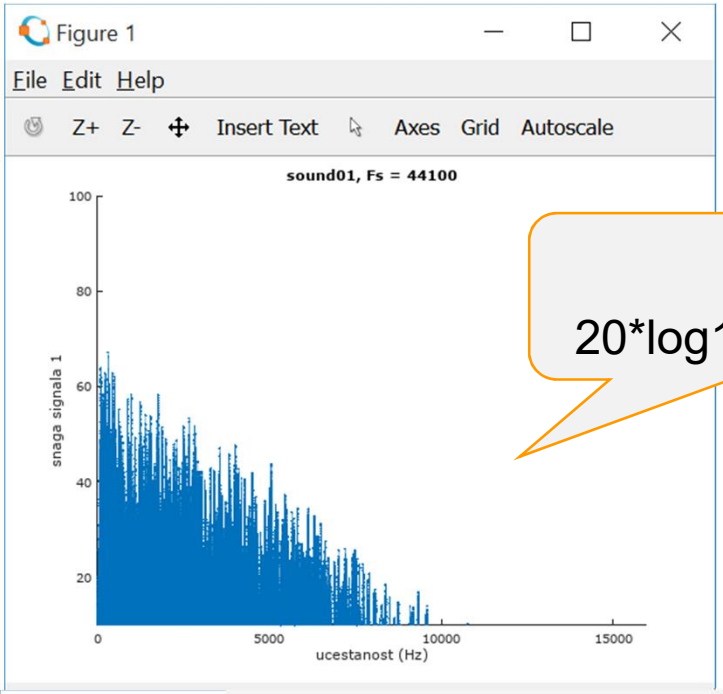
20*log10(abs(x1));





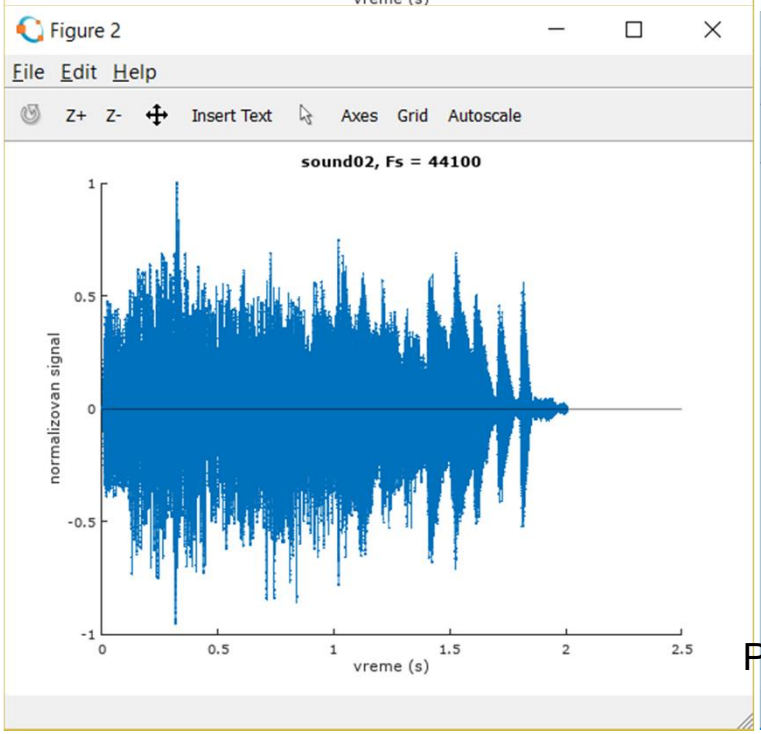


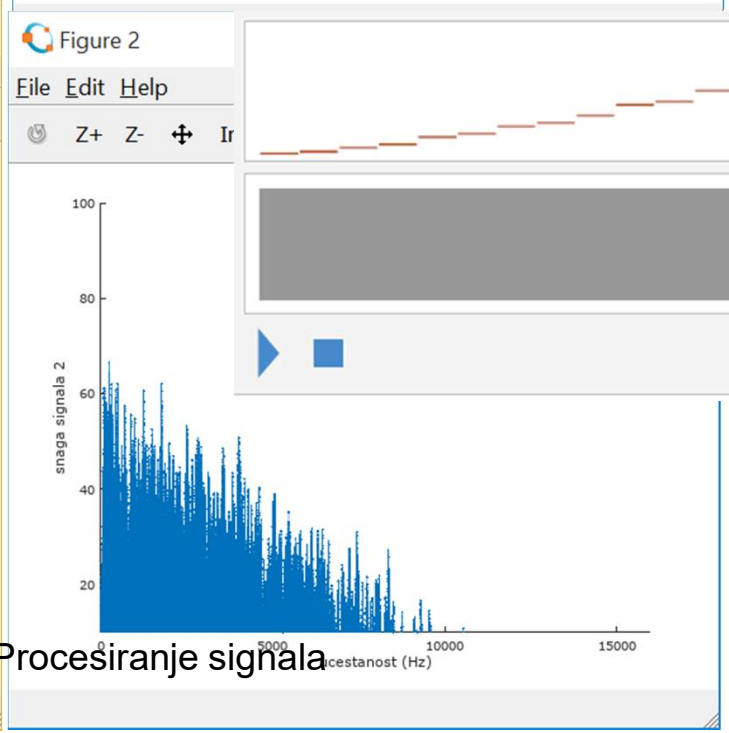


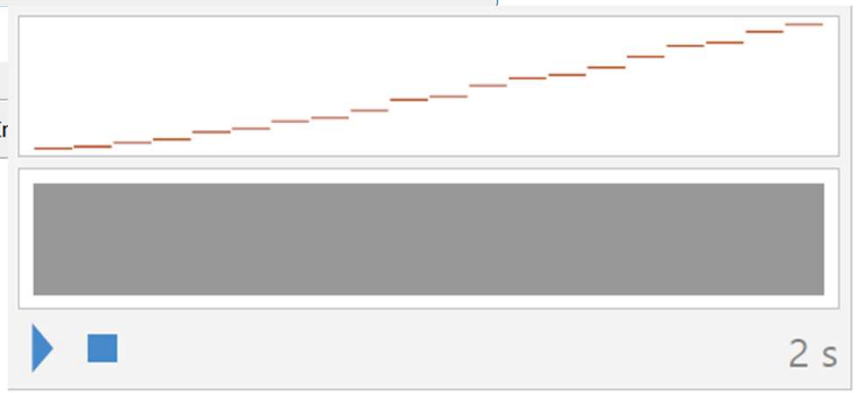


fft

20*log10(abs(x1));







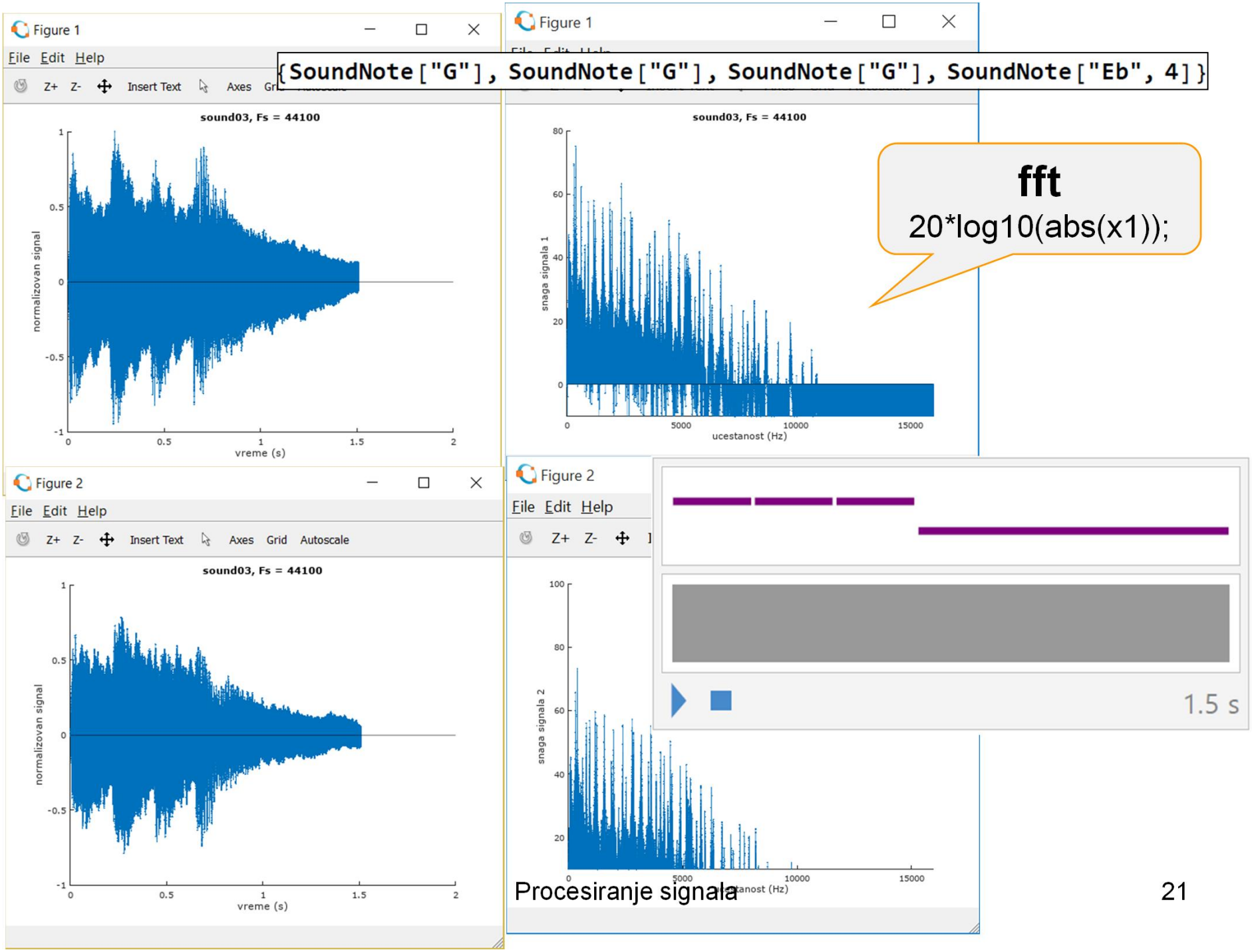
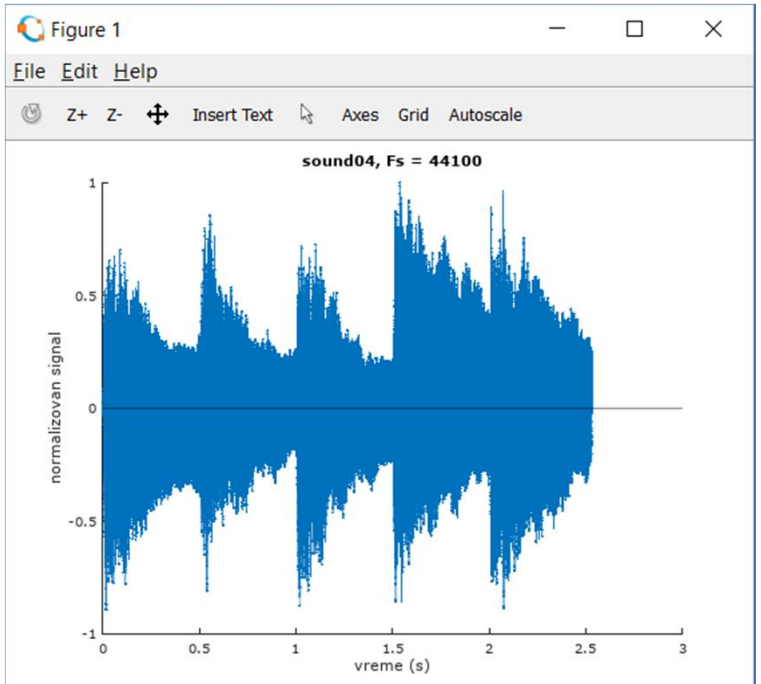
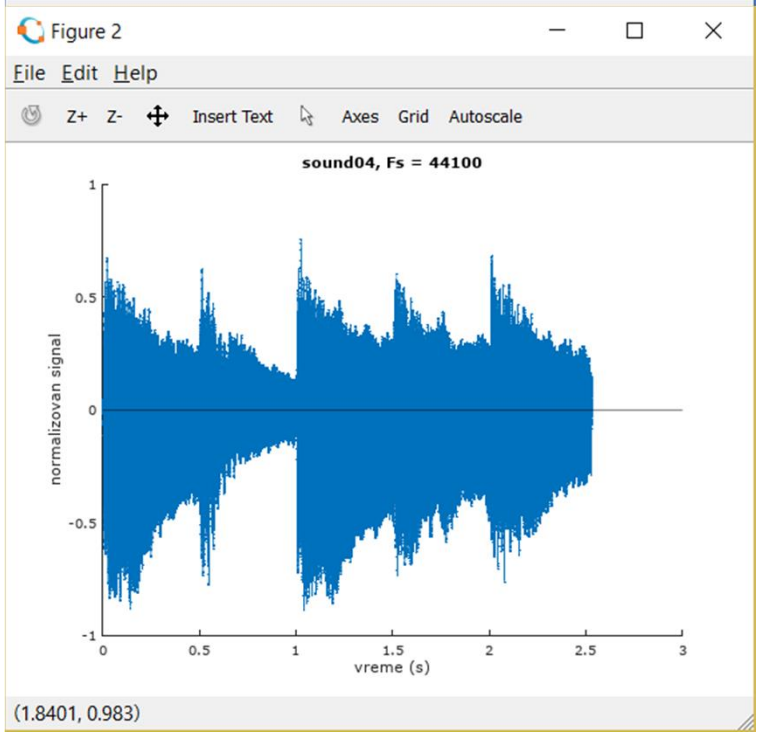
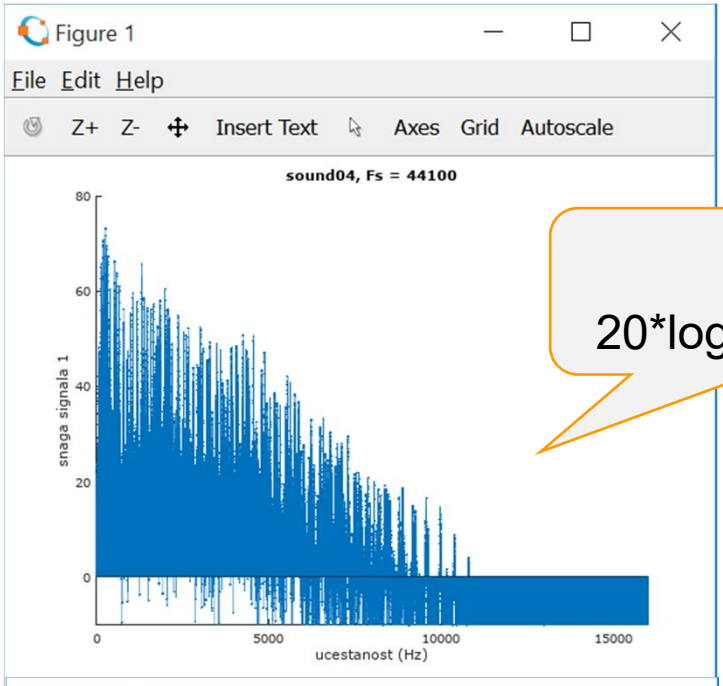


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound03, Fs = 44100
normalizovan signal
vreme (s)
{SoundNote["G"], SoundNote["G"], SoundNote["G"], SoundNote["Eb", 4]}
Figure 1
sound03, Fs = 44100
snaga signala 1
ucestanost (Hz)
fft
20*log10(abs(x1));
Figure 2
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound03, Fs = 44100
normalizovan signal
vreme (s)
Figure 2
File Edit Help
Z+ Z-
snaga signala 2
ucestanost (Hz)
1.5 s

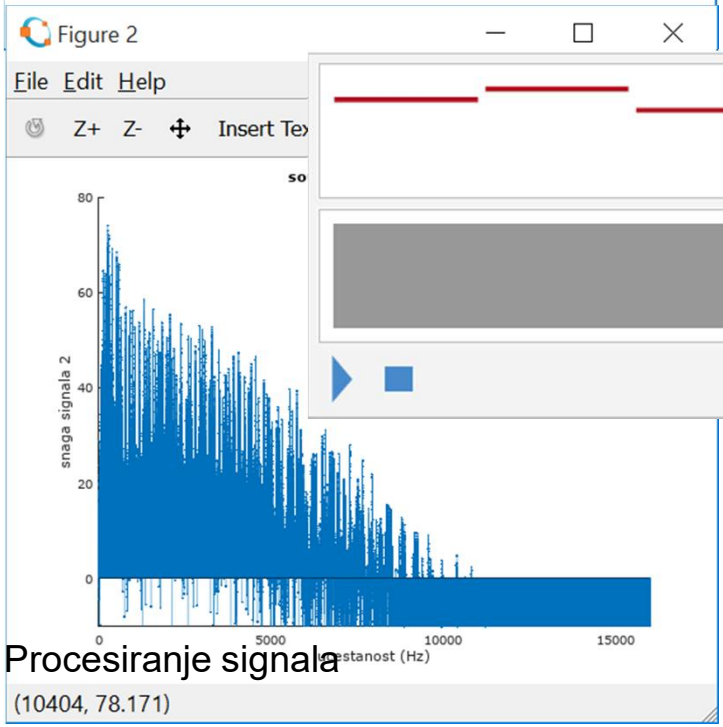


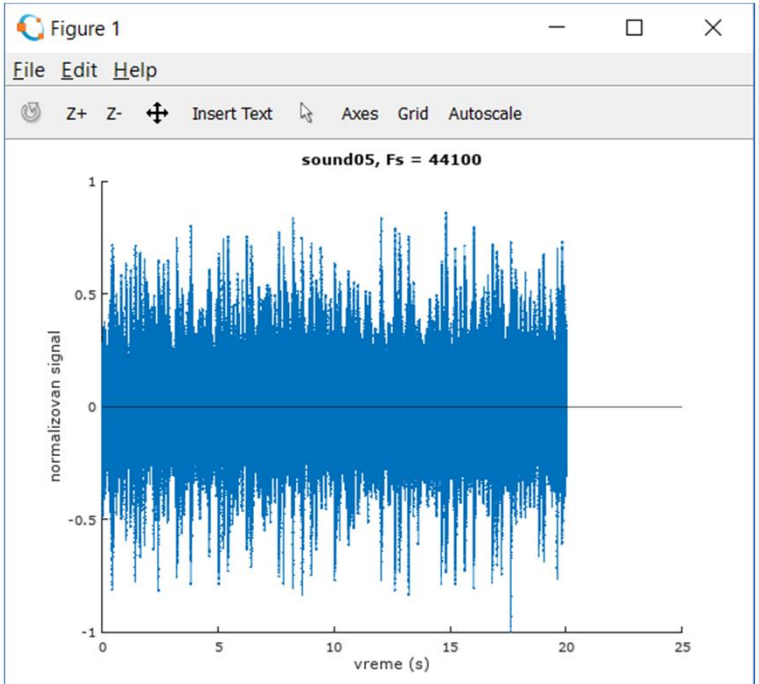


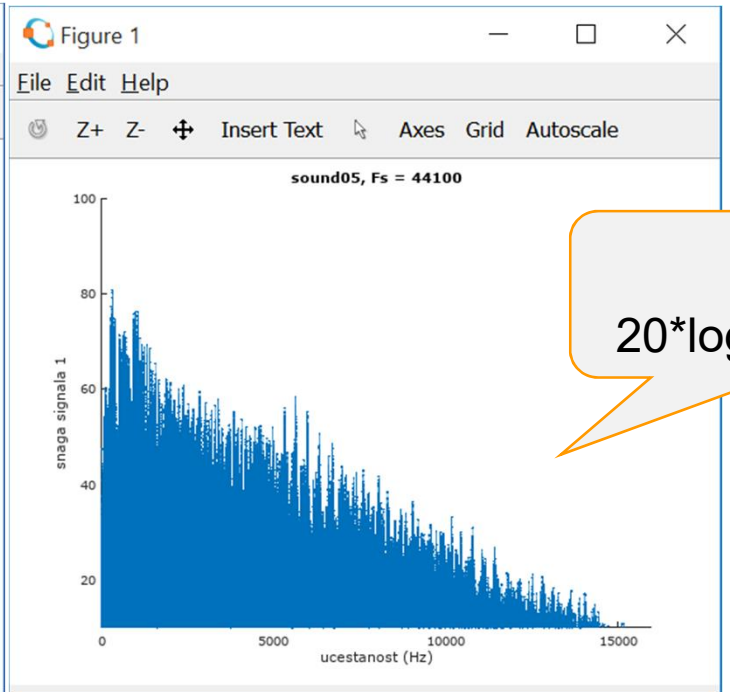


fft

20*log10(abs(x1));

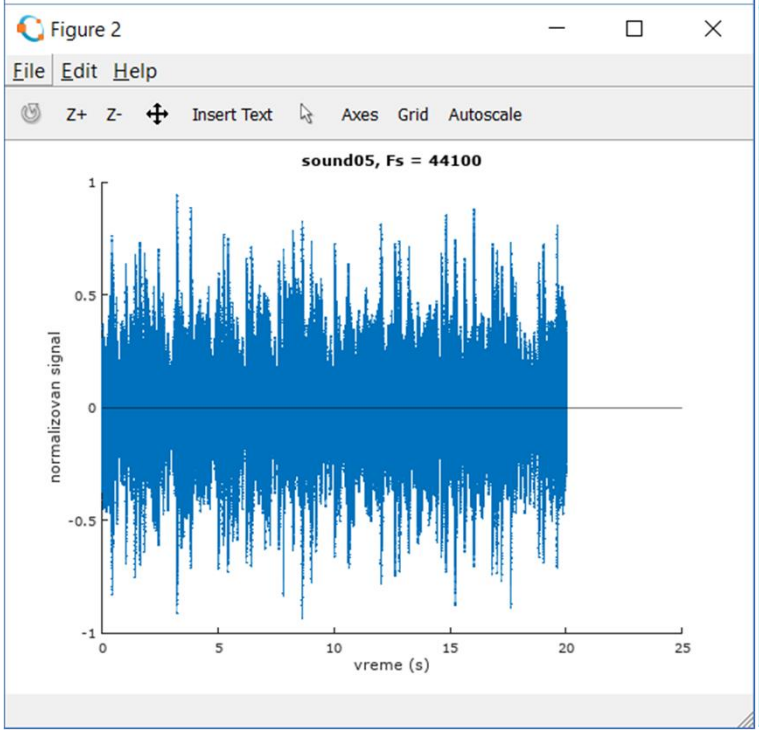


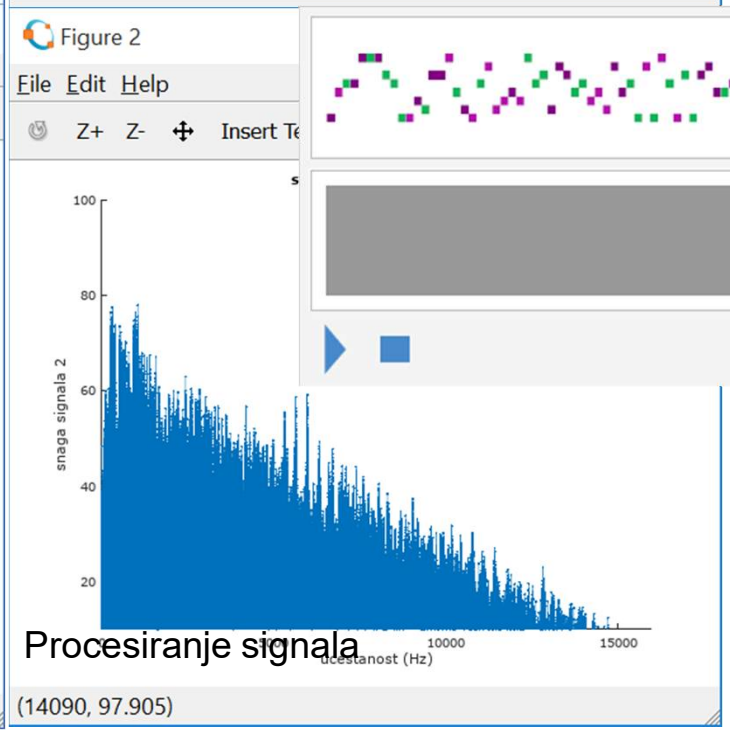


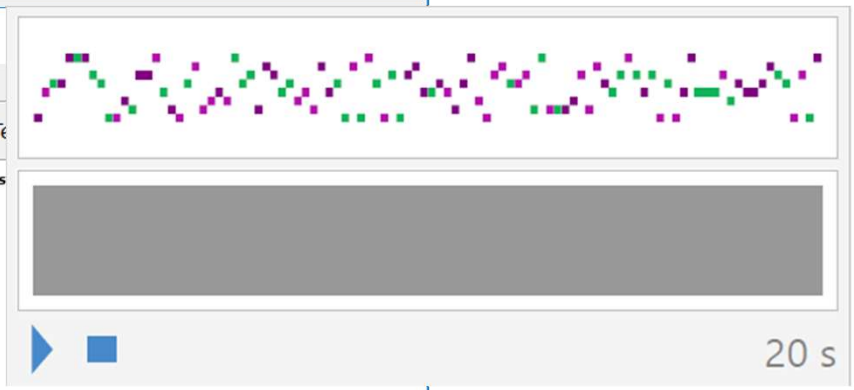


fft

20*log10(abs(x1));







pkg load signal

```
clear all, close all, clc
[y,fs]=audioread ("wav/sound01.wav");
y1=y(:,1);
y2=y(:,2);
nfft = 256;
ovlap = 32;
specgram(y1,nfft,fs,hamming(nfft),ovlap)
```

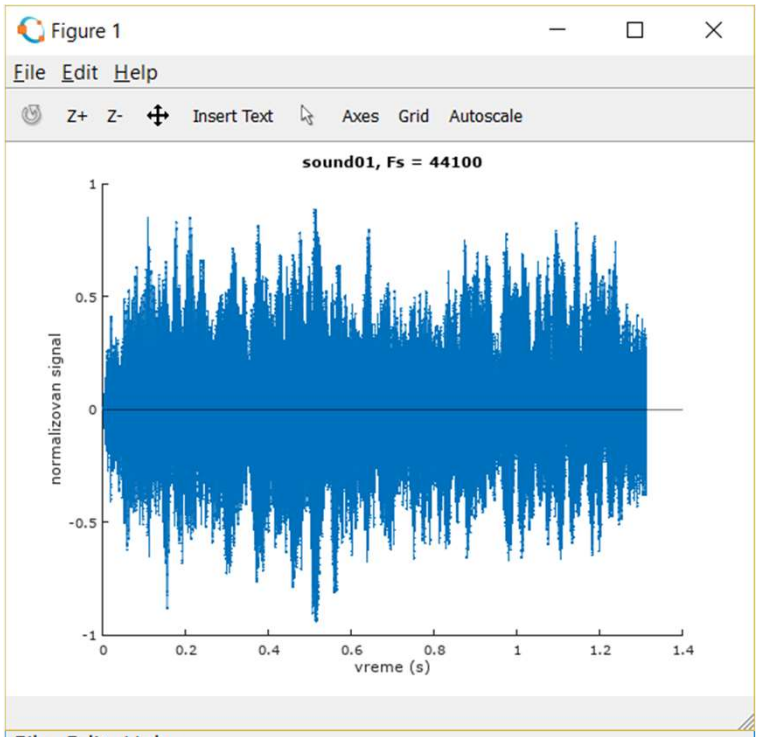
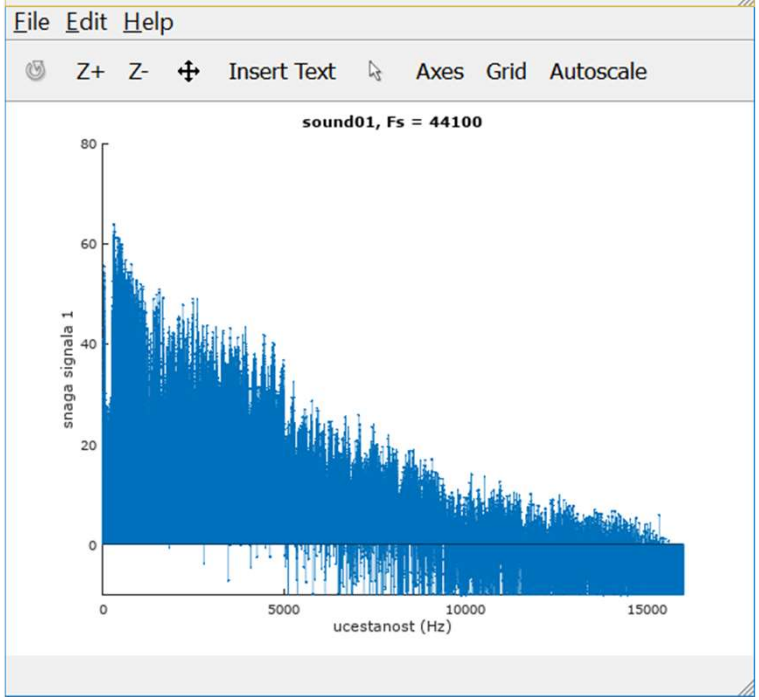


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound01, Fs = 44100
normalizovan signal
vreme (s)


File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound01, Fs = 44100
snaga signala 1
ucestanost (Hz)

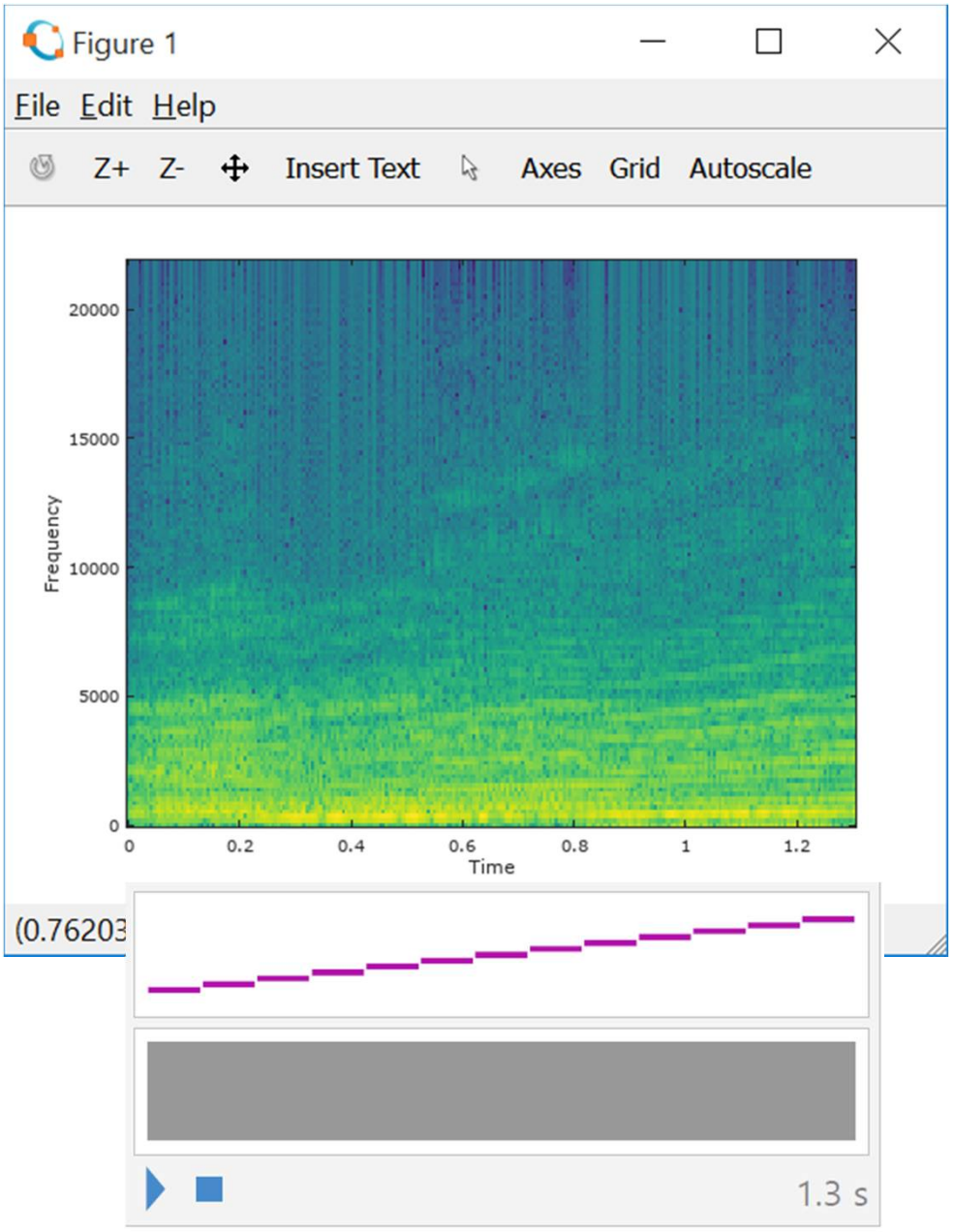


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Frequency
Time
(0.76203
1.3 s

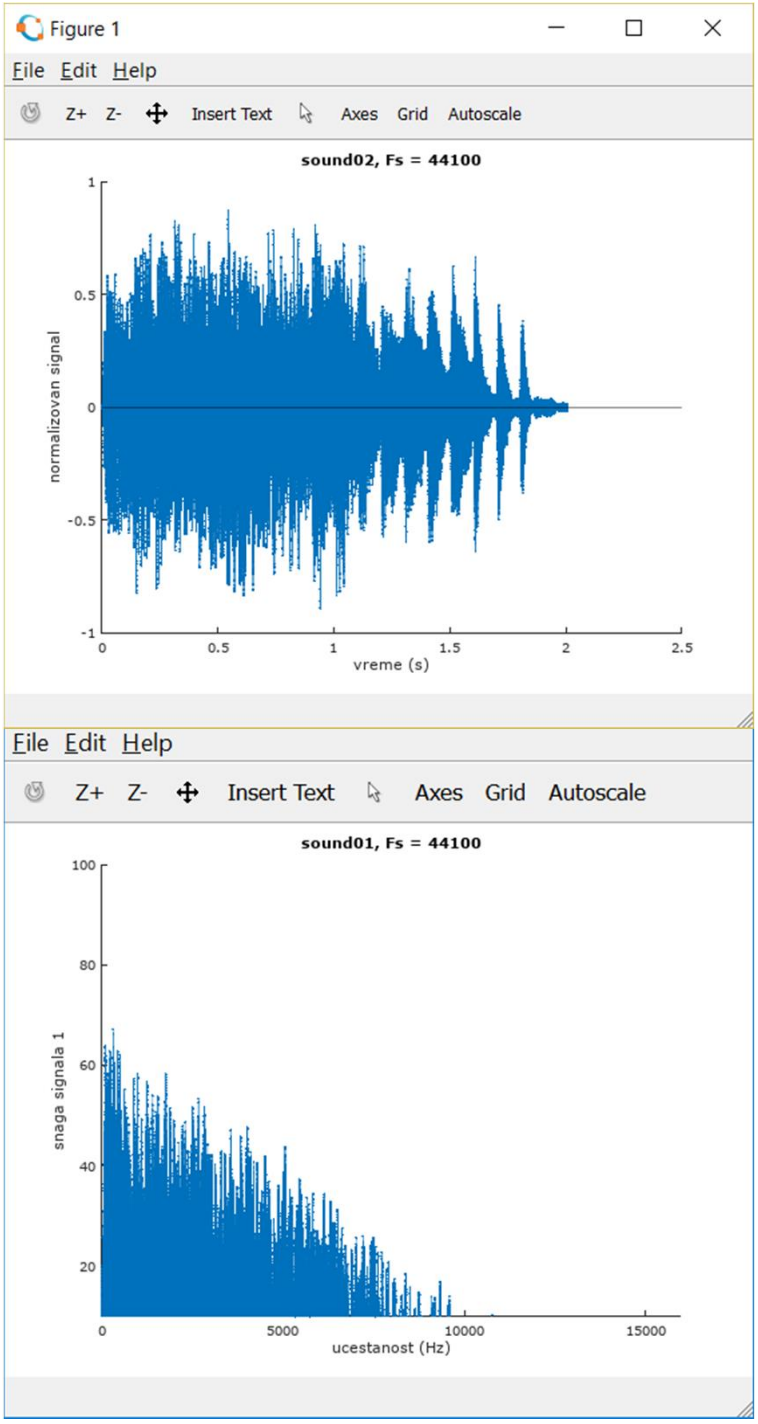


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound02, Fs = 44100
normalizovan signal
vreme (s)
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound01, Fs = 44100
snaga signala 1
ucestanost (Hz)

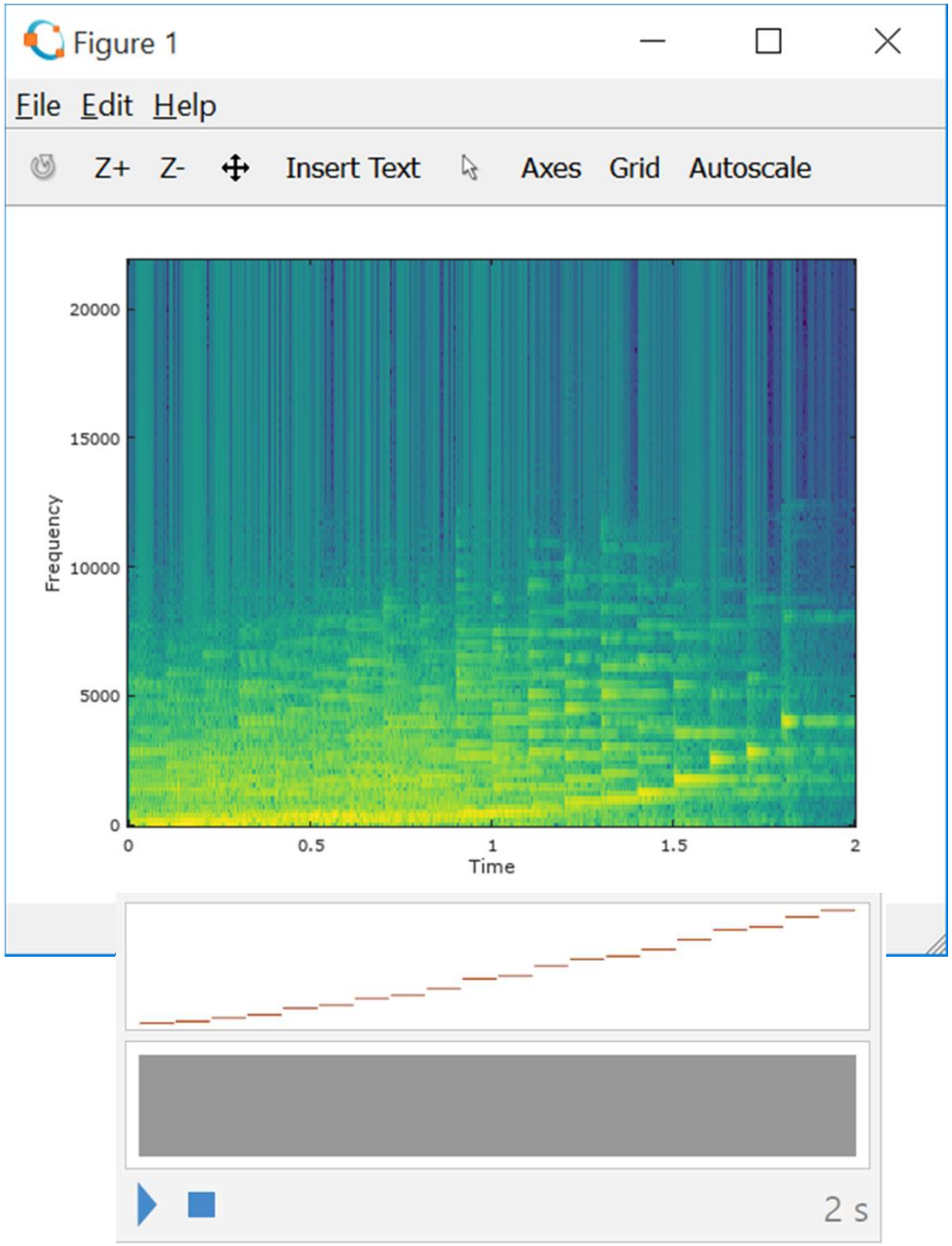


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Frequency
Time
2 s

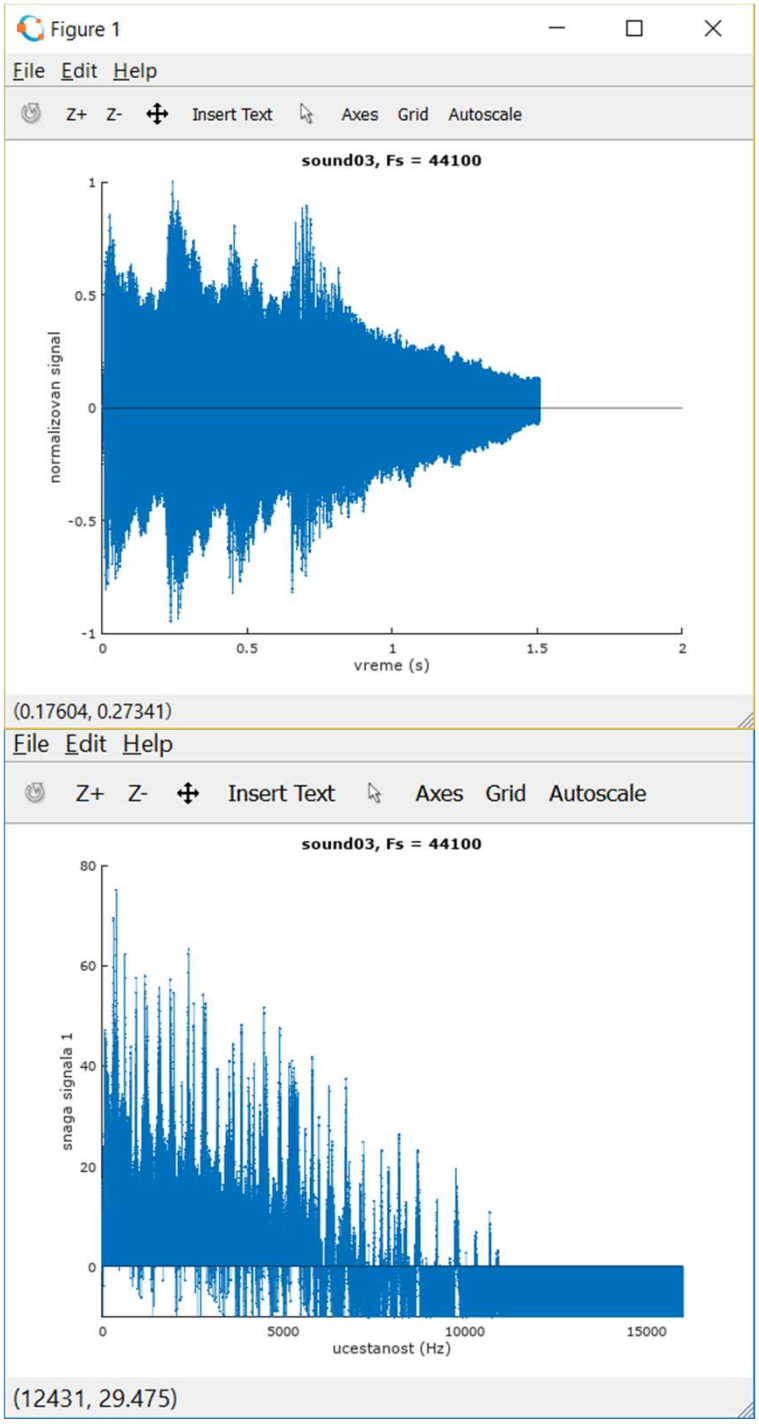


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound03, Fs = 44100
normalizovan signal
vreme (s)
(0.17604, 0.27341)
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound03, Fs = 44100
snaga signala 1
ucestanost (Hz)
(12431, 29.475)

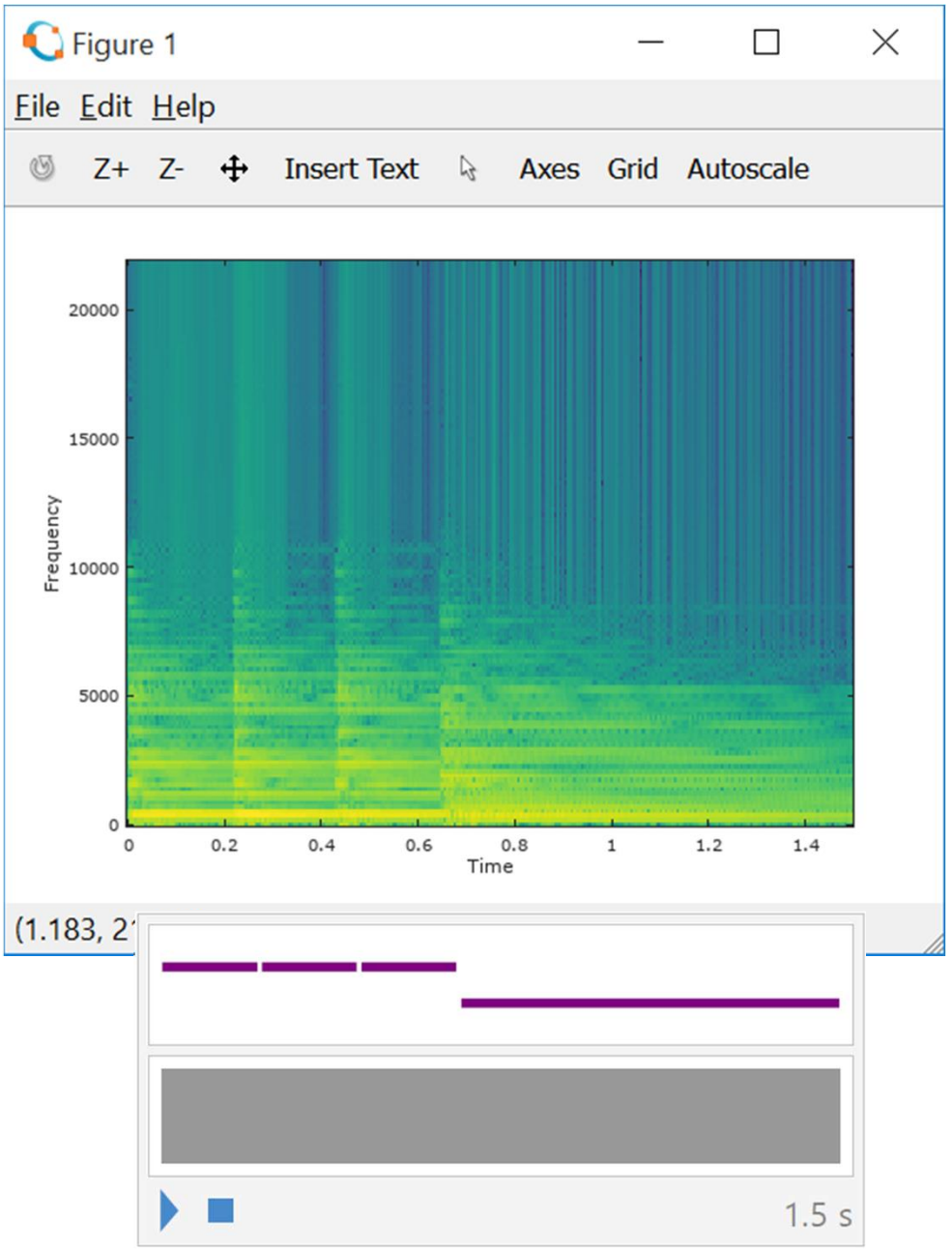


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Frequency
Time
1.5 s

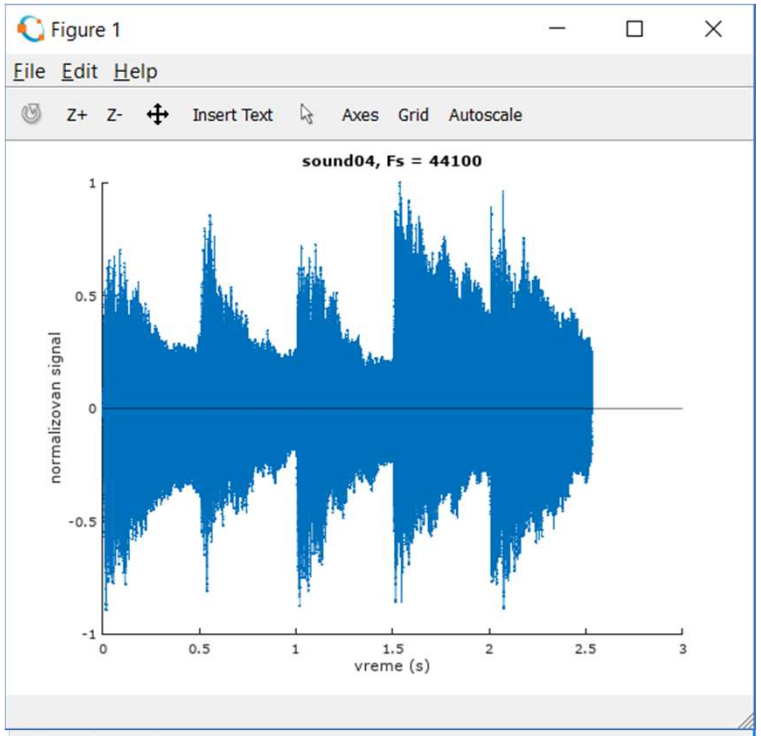
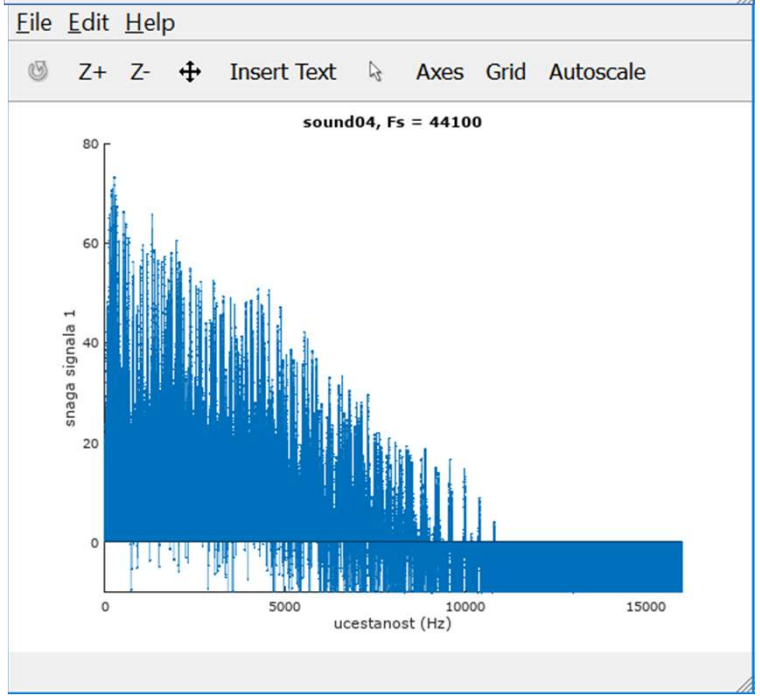


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound04, Fs = 44100
normalizovan signal
vreme (s)


File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound04, Fs = 44100
snaga signala 1
ucestanost (Hz)

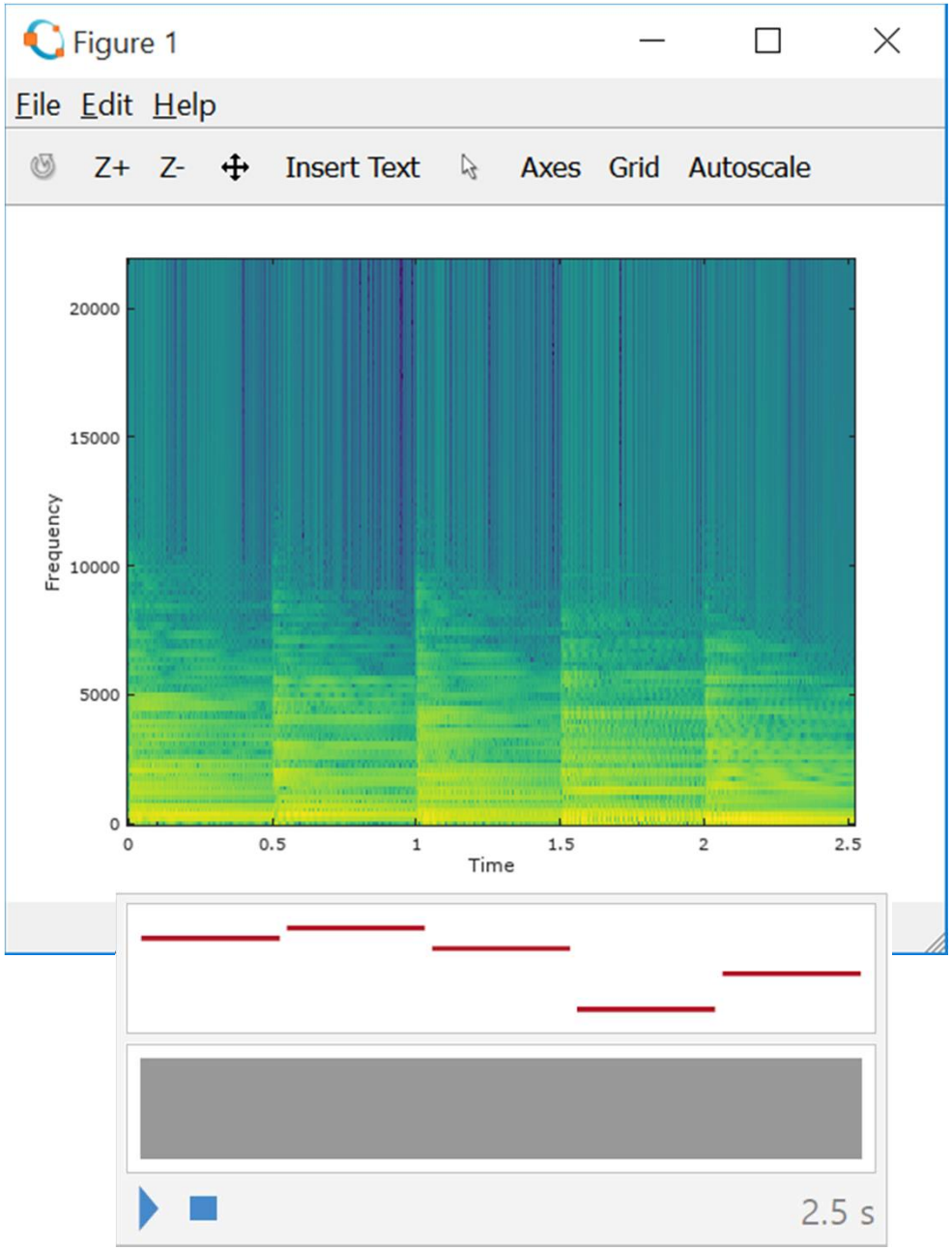


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Frequency
Time
2.5 s

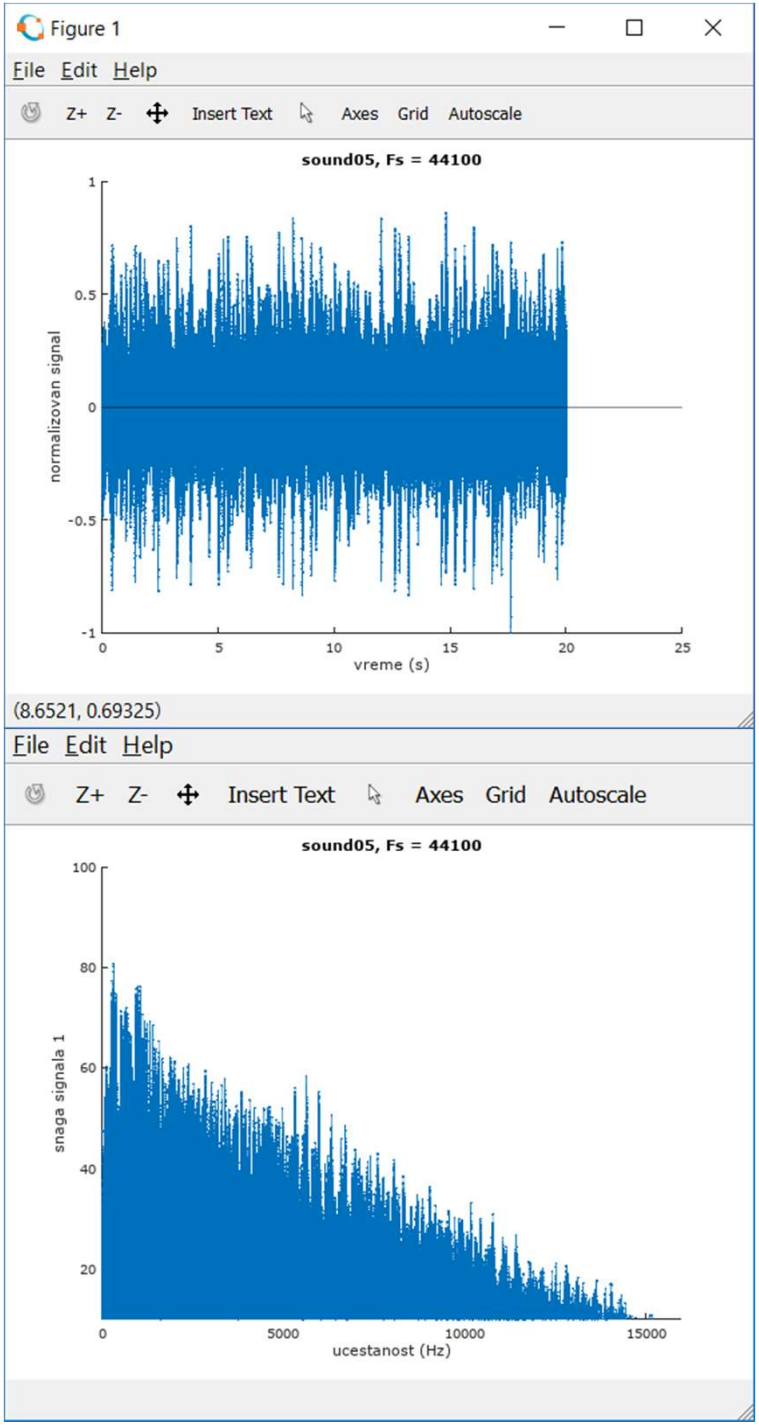


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound05, Fs = 44100
normalizovan signal
vreme (s)
(8.6521, 0.69325)
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
sound05, Fs = 44100
snaga signala 1
ucestanost (Hz)

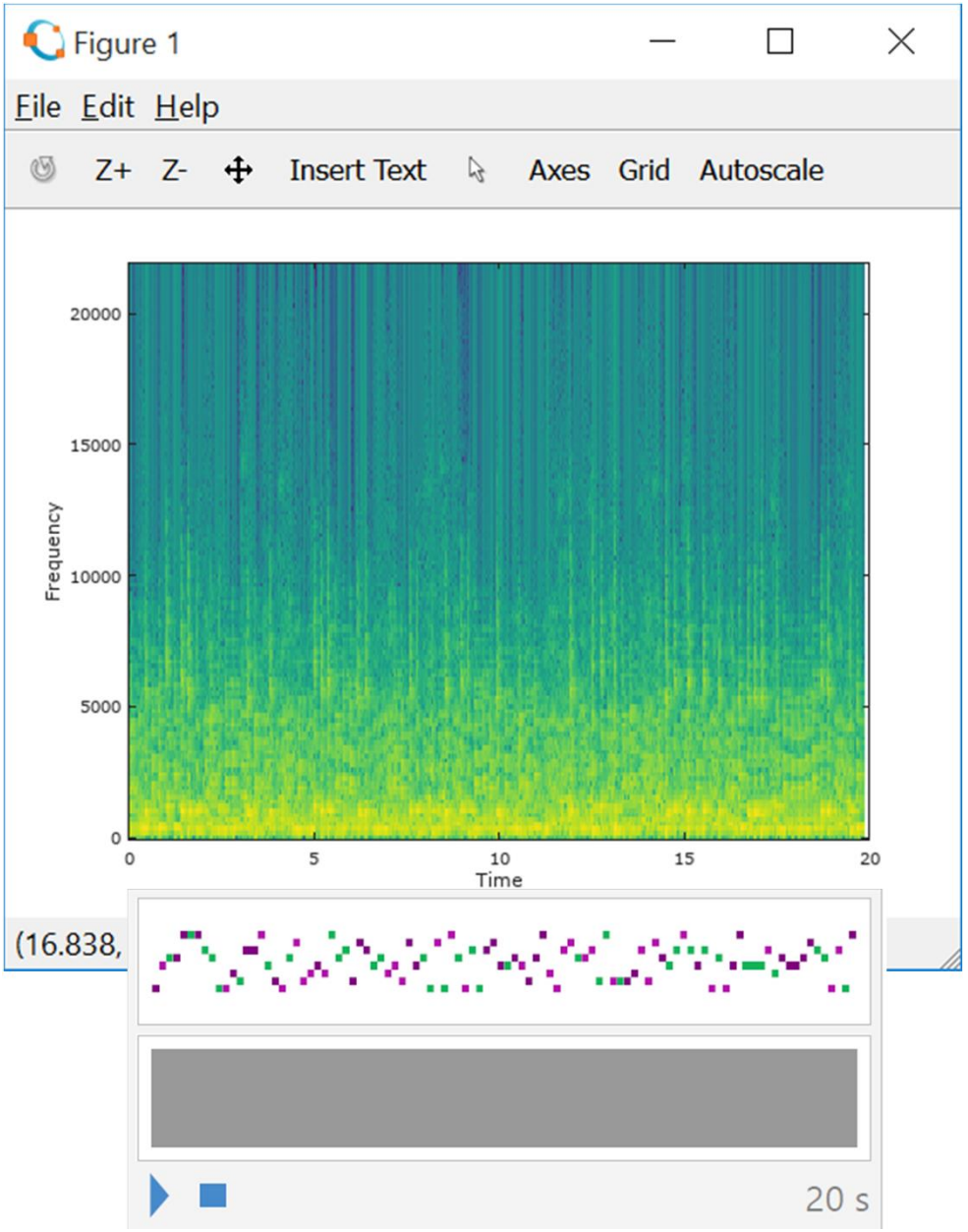
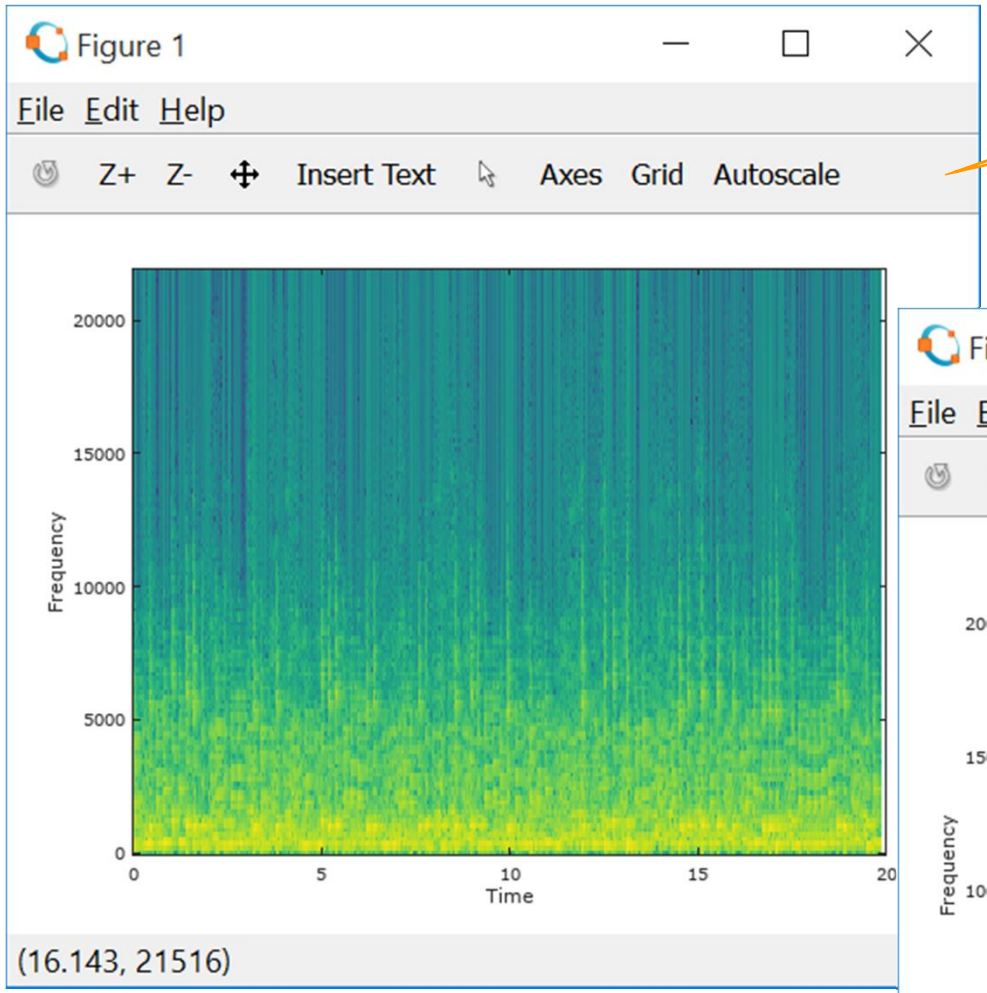
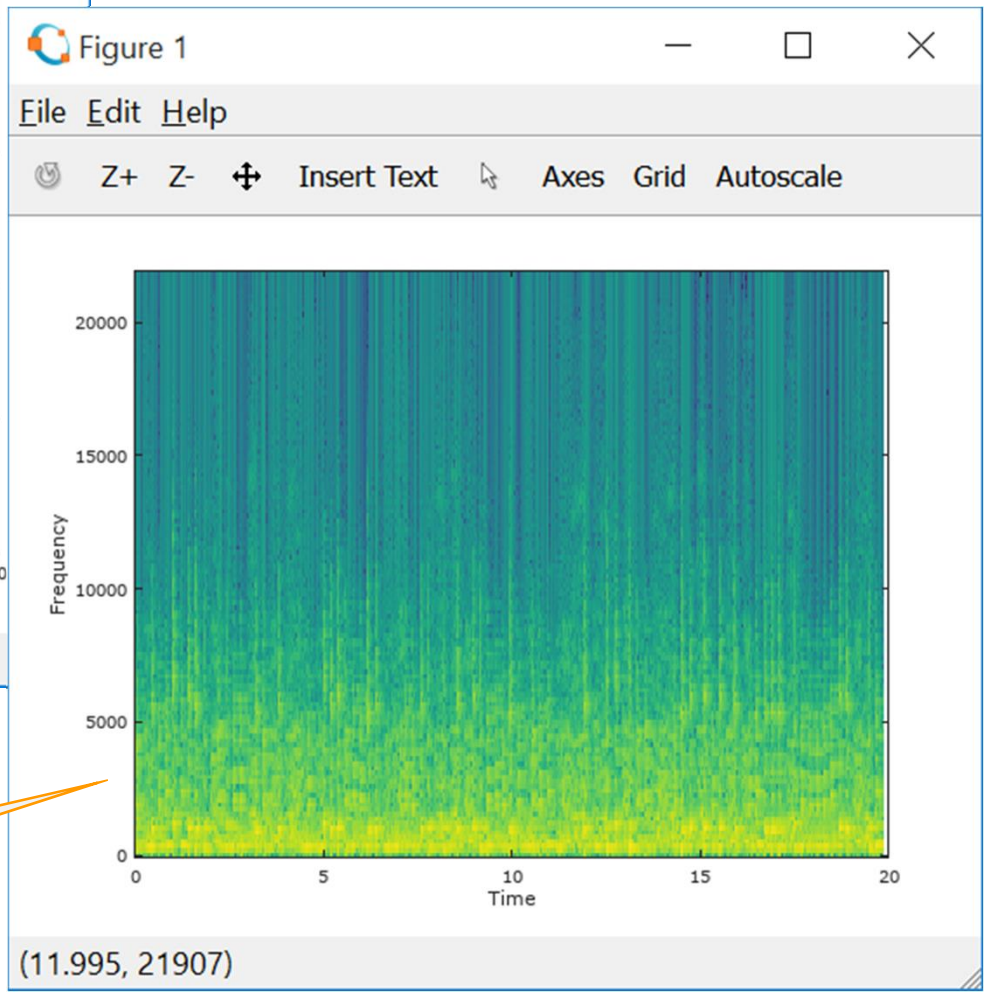


Figure 1
File Edit Help
Z+ Z- Insert Text Axes Grid Autoscale
Frequency
Time
(16.838,
20 s



ovlap = 0;



ovlap = 128;

Diskretna Furijeova transformacija

Sekvenca u vremenskom domenu $x[n]$ dužine N preslikava se u sekvencu $X[k]$, iste dužine gde se članovi računaju po formuli

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}, \quad 0 \le k \le N-1$$

Matlab `fft` realizuje DFT

`X=fft[x]`

- Iz zadate sekvence u vremenskom domenu $x[n]$ dobija se nova sekvenca u $X[k]$

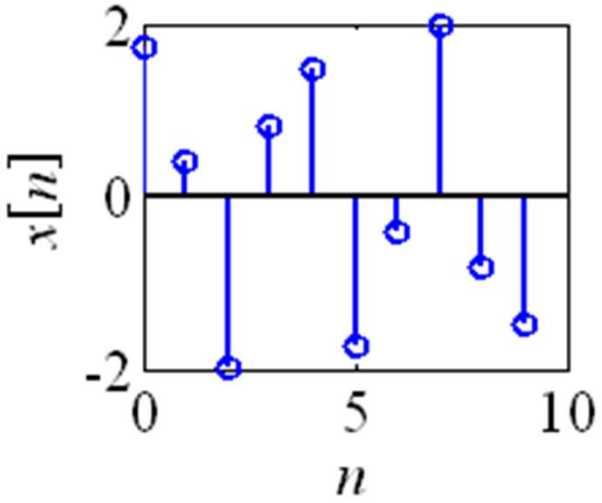
```
x = [1 1/2 1/4 1/8 1/16 1/32 1/64 1/128]
X = fft(x)
```

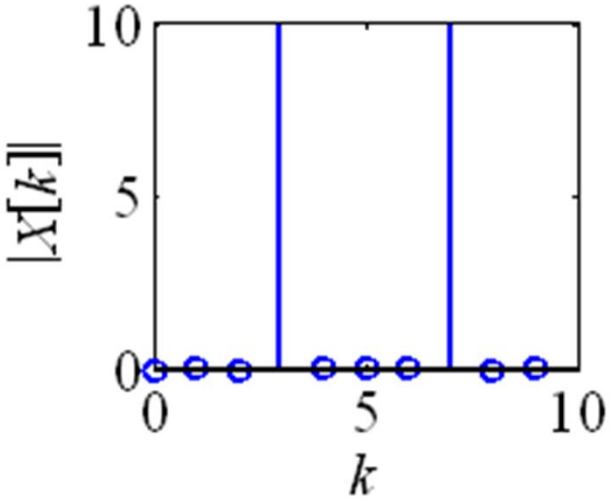
Kompleksni brojevi

```
x =
 1.0000 0.5000 0.2500 0.1250 0.0625 0.0313 0.0156 0.0078
```

```
X =
  Columns 1 through 4
   1.9922  1.1861-0.6487i 0.7969-0.3984i 0.6889-0.1799i
  Columns 5 through 8
   0.6641  0.6889+0.1799i 0.7969+0.3984i 1.1861+0.6487i
```

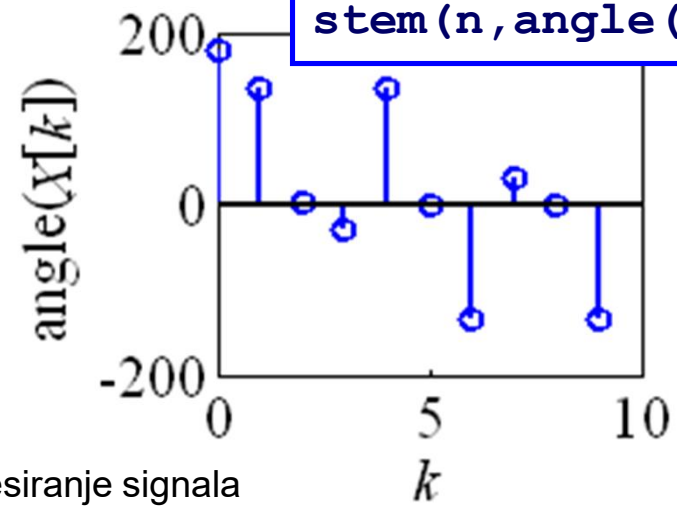
FFT



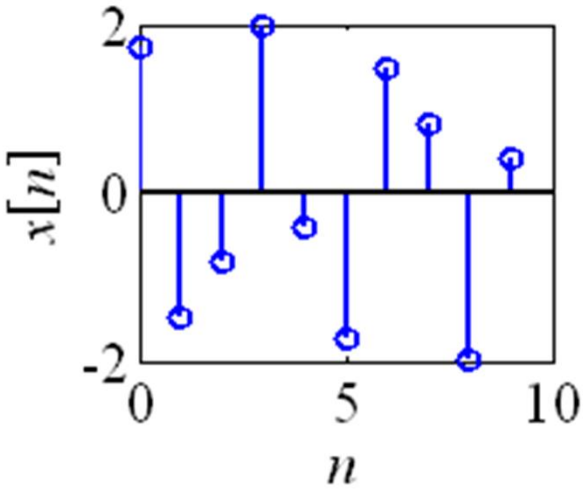


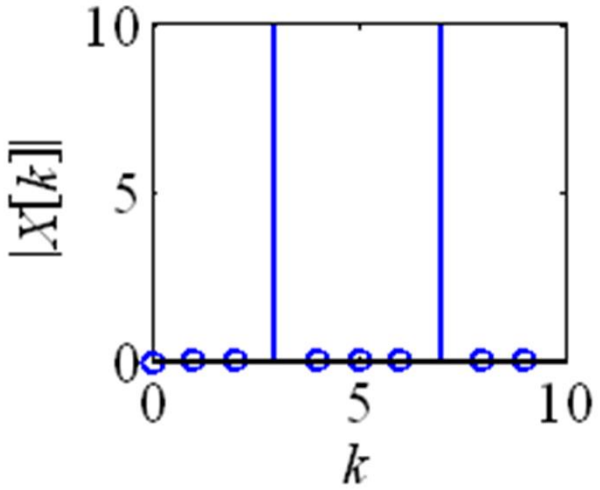

```
N = 10;
n = 0:N-1;
k = 3;
f = k/N;
fi = -pi/6;
x = 2*cos(2*pi*f*n+fi);
stem(n,x)
X = fft(x)
stem(n,abs(X))
```

```
stem(n,angle(X)*180/pi)
```

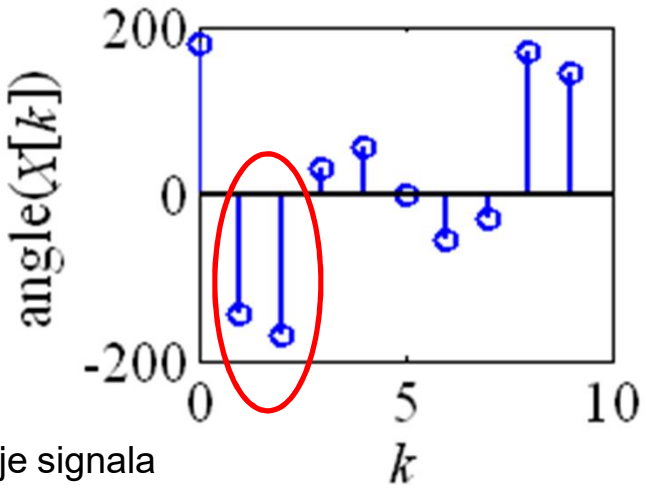


FFT - promena faze

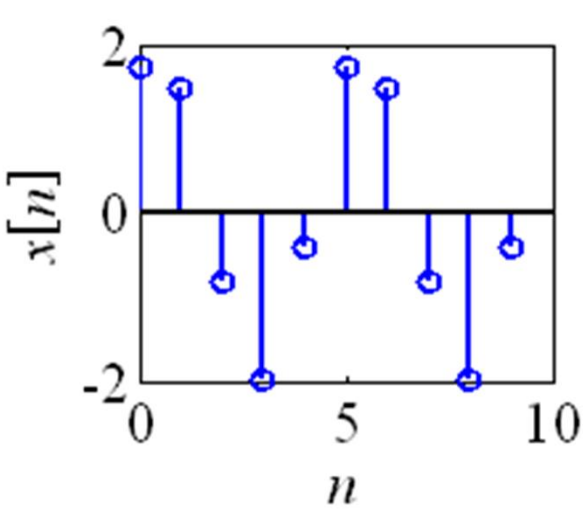


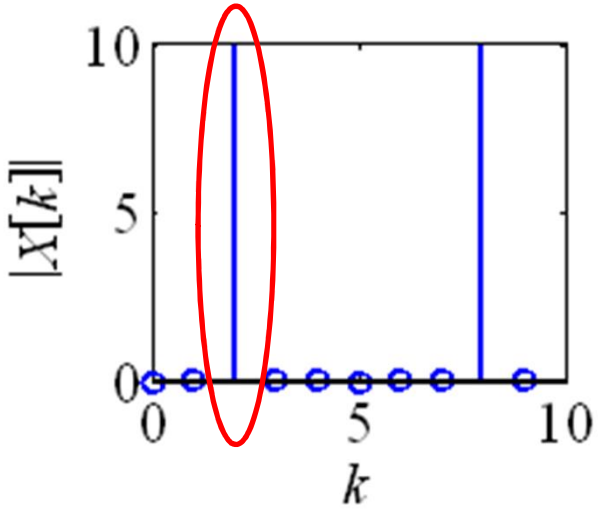



```
N = 10;
n = 0:N-1;
k = 3;
f = k/N;
fi = pi/6;
x = 2*cos(2*pi*f*n+fi);
stem(n,x)
X = fft(x)
stem(n,abs(X))
```

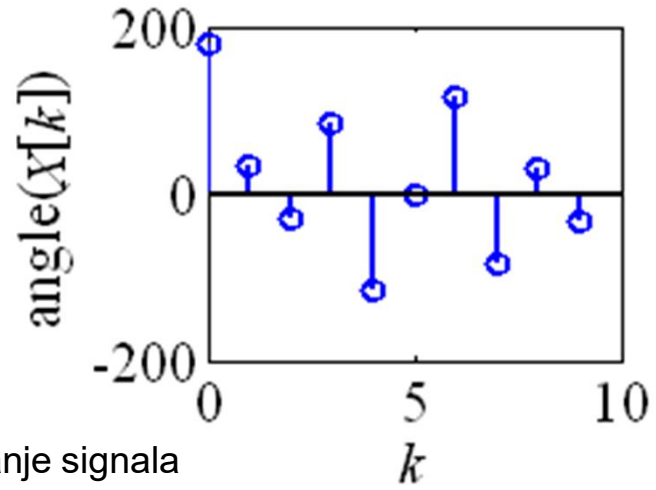


FFT – promena *k*

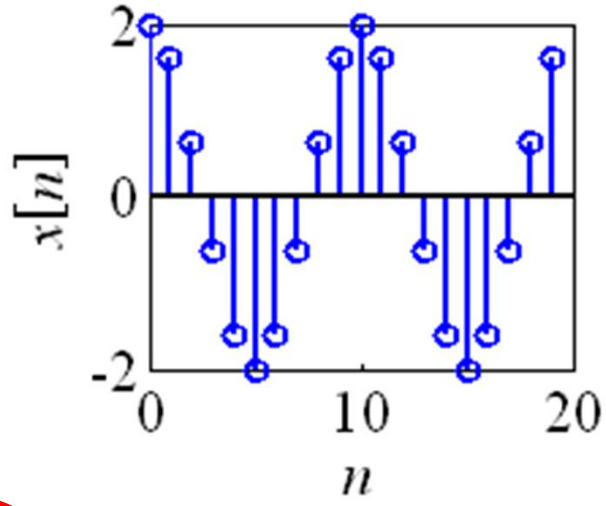


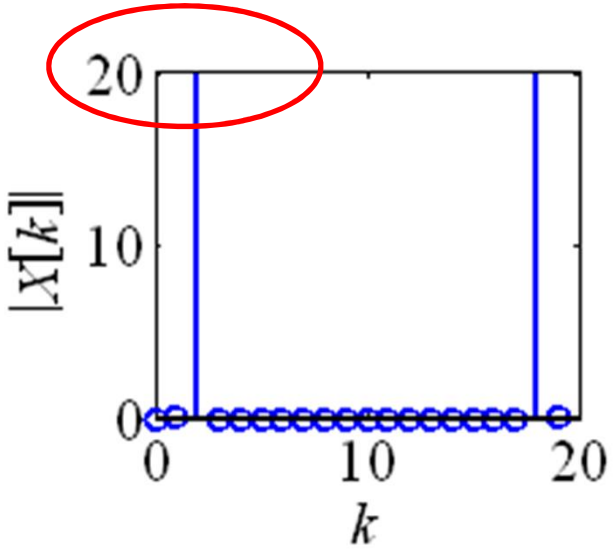



```
N = 10;
n = 0:N-1;
k = 2;
f = k/N;
fi = -pi/6;
x = 2*cos(2*pi*f*n+fi);
stem(n,x)
X = fft(x)
stem(n,abs(X))
```

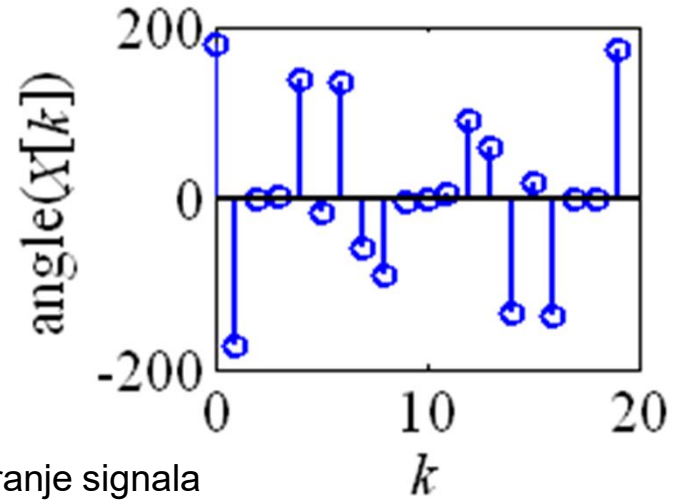


FFT – promena *N*

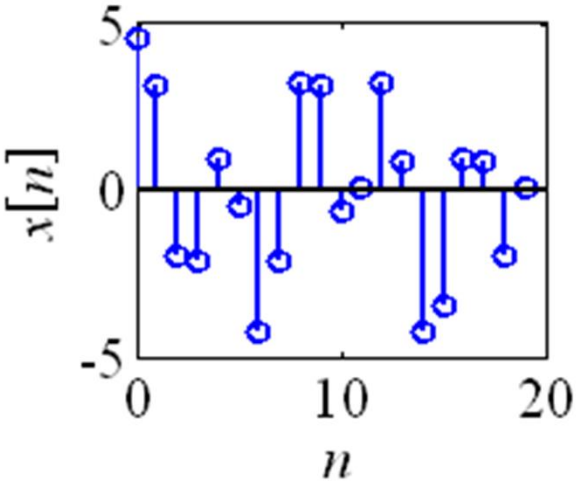


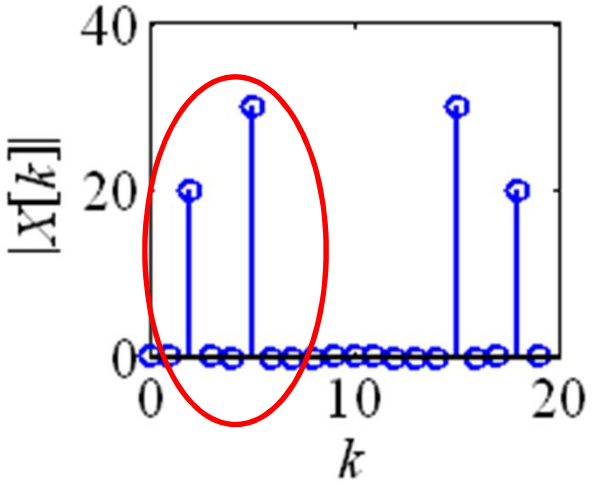



```
N = 20;
n = 0:N-1;
k = 3;
f = k/N;
fi = -pi/6;
x = 2*cos(2*pi*f*n+fi);
stem(n,x)
X = fft(x)
stem(n,abs(X))
```

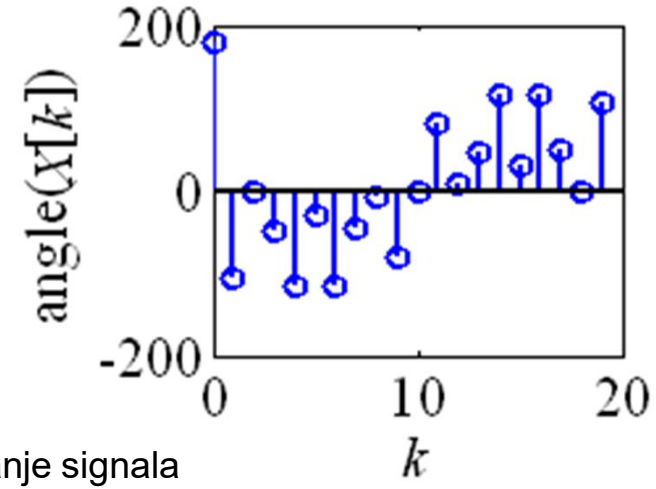


FFT – dve sinusoide



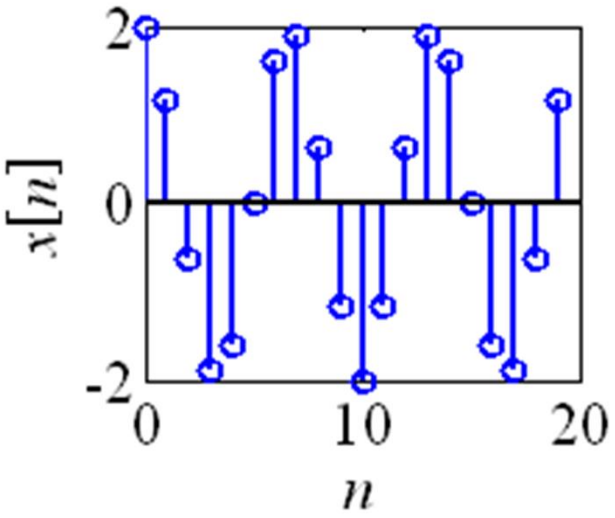


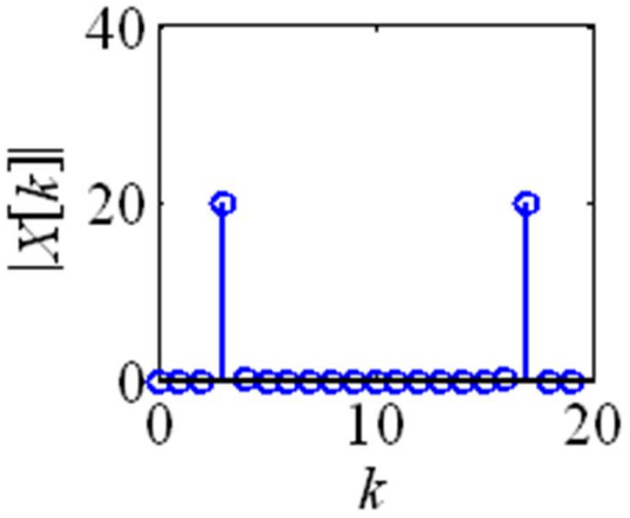

```
N=20;
n=0:N-1;
k1=2;     k2=5;
f1=k1/N;  f2=k2/N;
fi2=-pi/6;
x = 2*cos(2*pi*f1*n+0)+...
    3*cos(2*pi*f2*n+fi2);
X = fft(x)
```



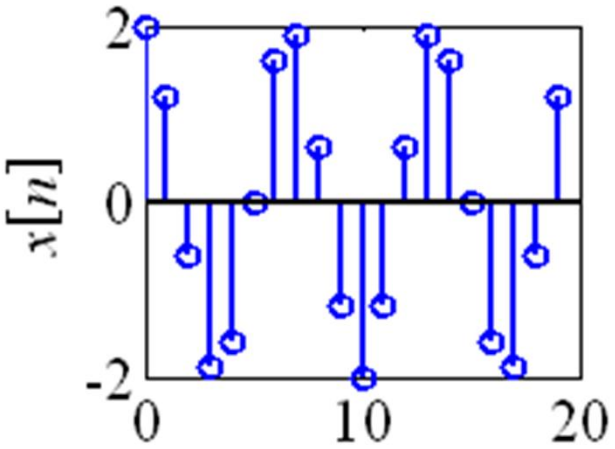
FFT - aliasing

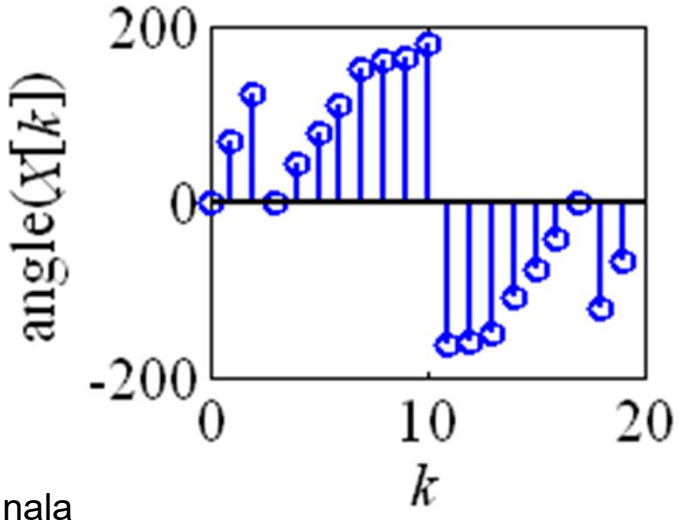
```
k=3;
f=k/N;
```






```
k=N+3;
f=k/N;
```

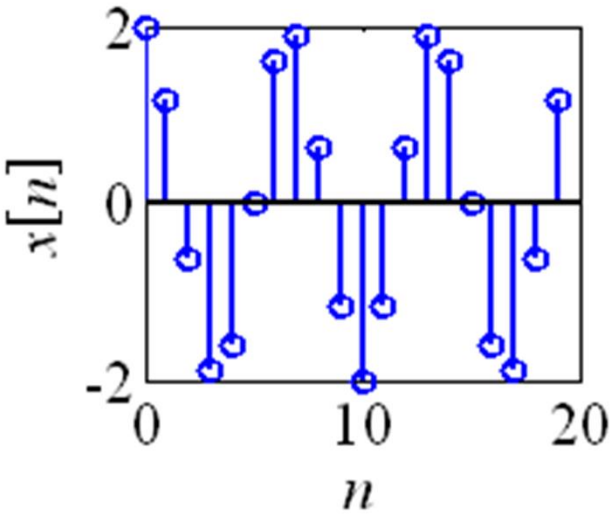


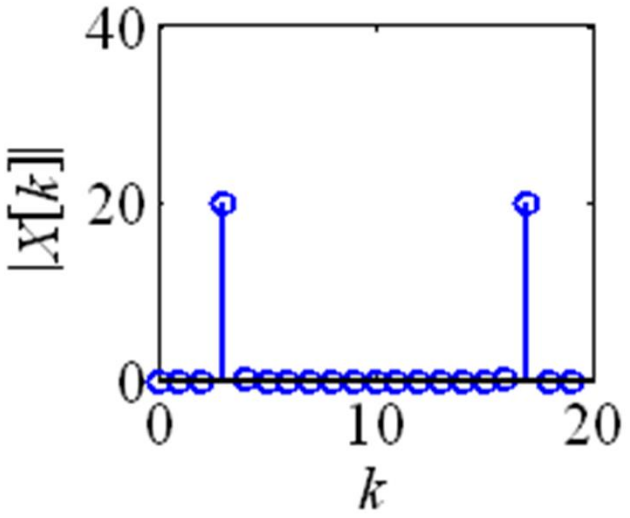



```
x = 2*cos(2*pi*f*n);
```

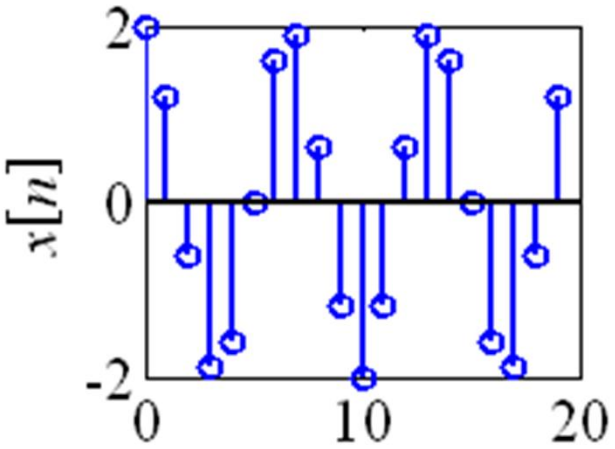
FFT - folding

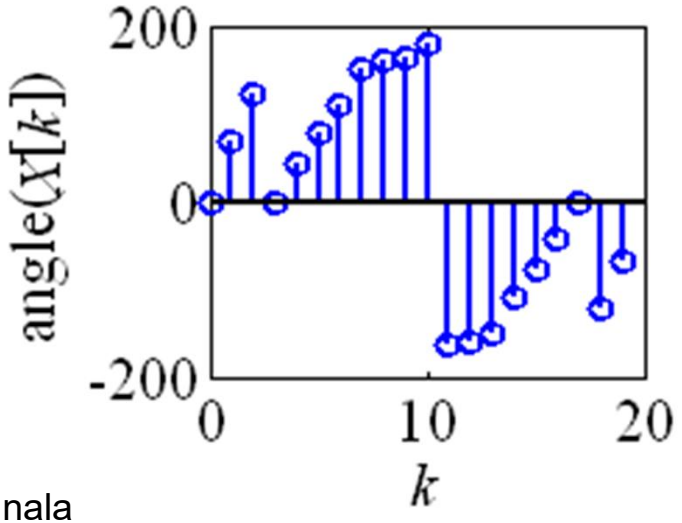
```
k=3;
f=k/N;
```






```
k=N-3;
f=k/N;
```






```
x = 2*cos(2*pi*f*n);
```

Jednoznačnost FFT

```
k=3;
f=k/N;
```

$$k < N/2$$

```
x = 2*cos(2*pi*f*n);
```

$$f < 1/2$$

Jednoznačnost FFT

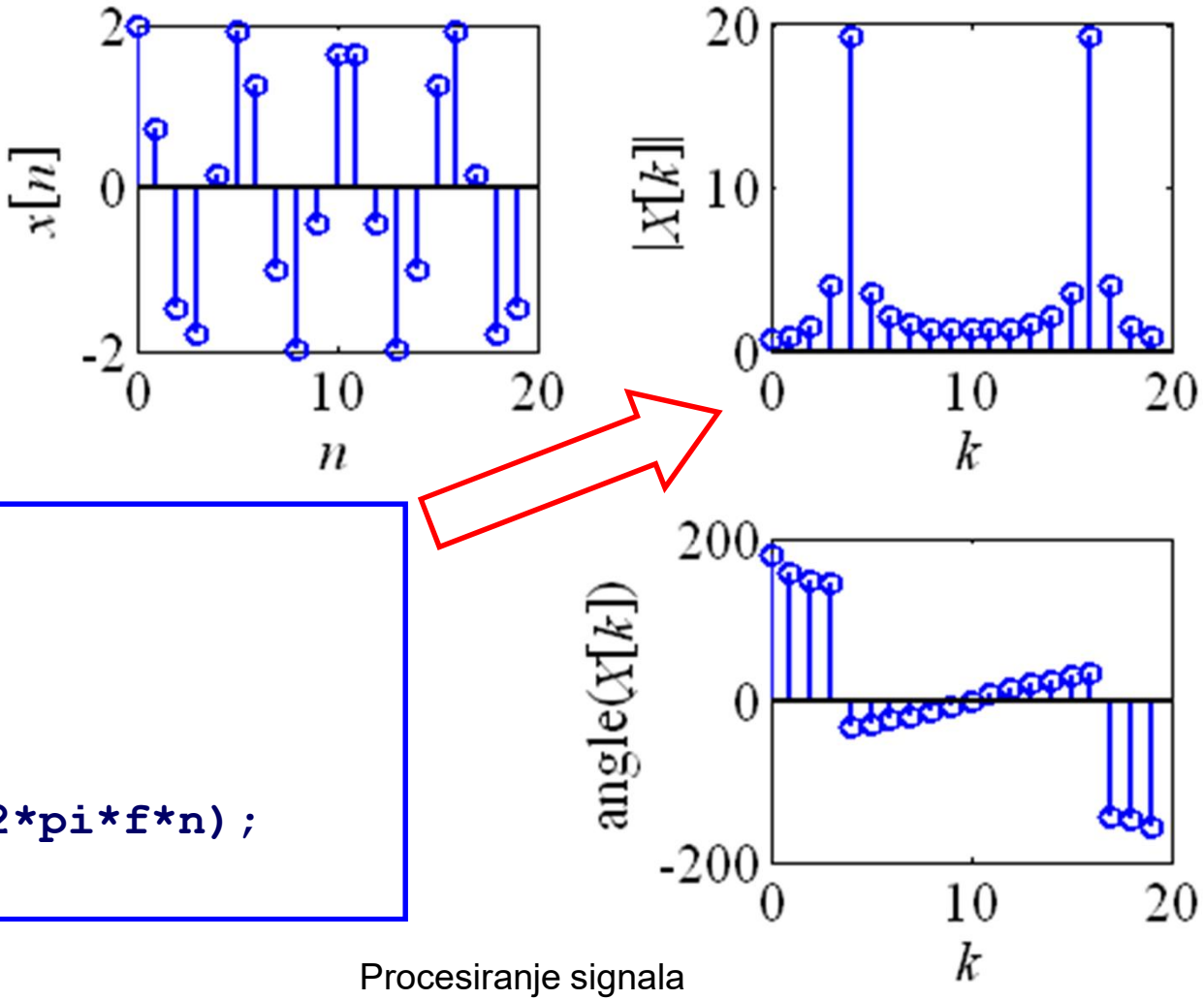
```
x = 2*cos(2*pi*f*n);
```

```
f < 1/2
f = 1-f
f = 1+f
f = 2-f
f = 2+f
```

FFT – curenje spektra

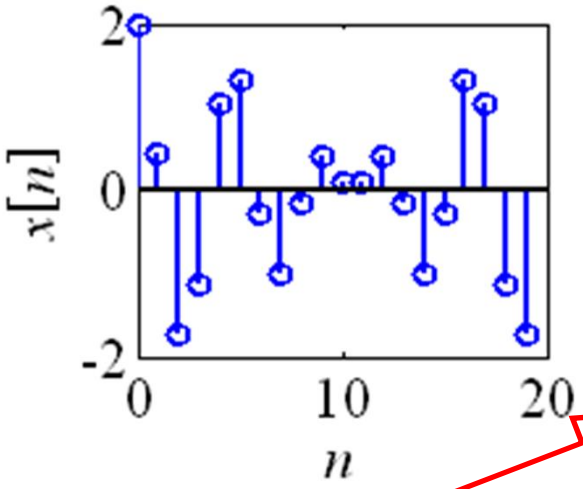
frequency leakage

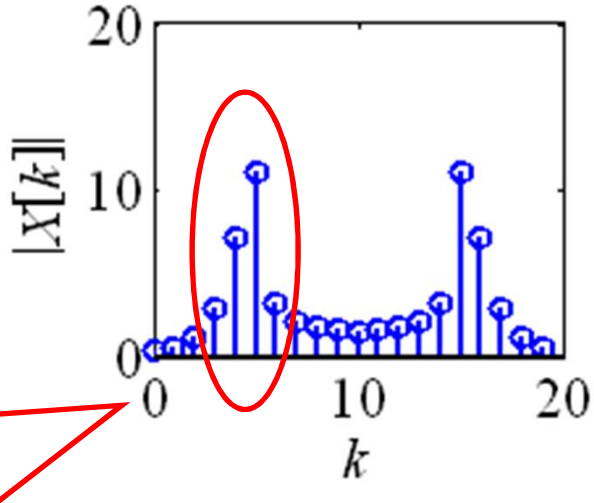
```
N = 20;
M = 21;
n = 0:N-1;
k = 4;
f = k/M;
x = 2*cos(2*pi*f*n);
X = fft(x)
```

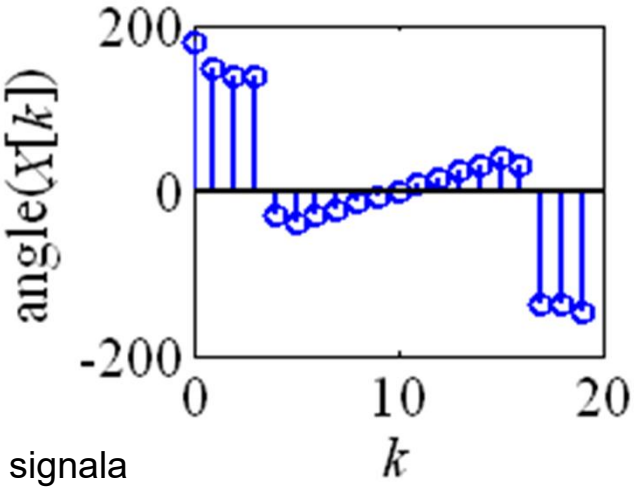


FFT – frekvencijska rezolucija

frequency resolution



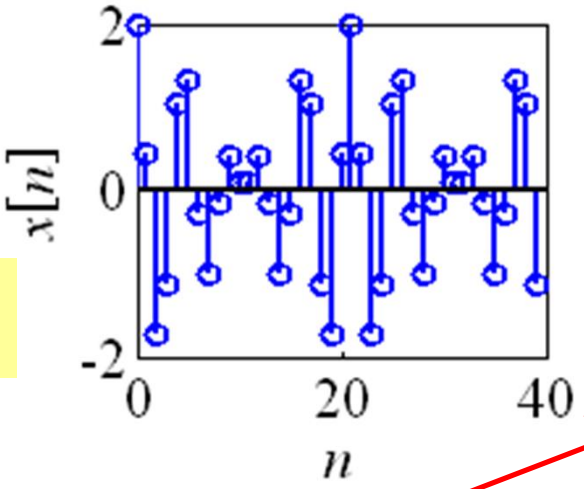


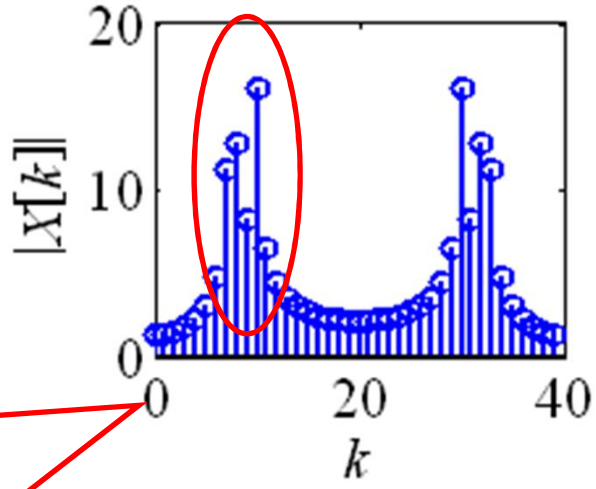


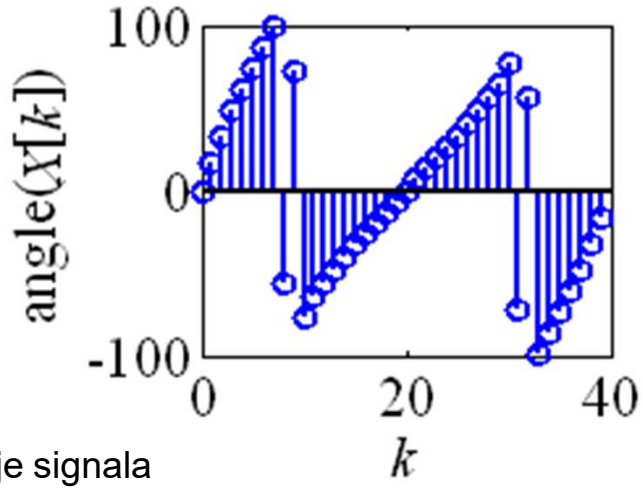

```
N = 20;
M = 21;
n = 0:N-1;
k1=4; f1=k1/M;
k2=5; f2=k2/M;
x=cos(2*pi*f1*n)+...
  cos(2*pi*f2*n);
X = fft(x)
```

FFT – frekvencijska rezolucija

$$\Delta f = 1/N$$

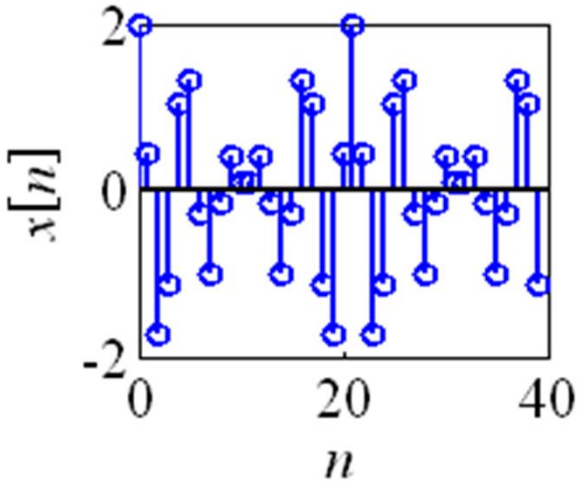


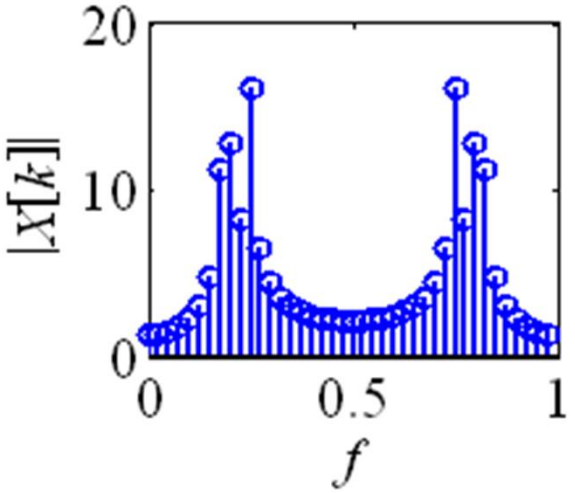




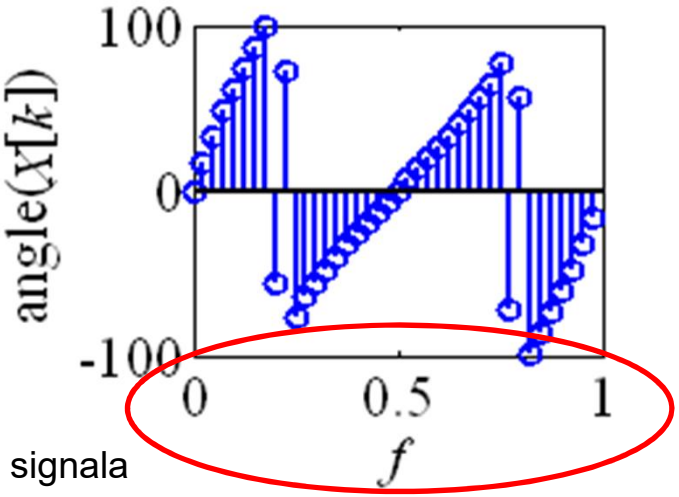

```
N = 40;
M = 21;
n = 0:N-1;
k1=4; f1=k1/M;
k2=5; f2=k2/M;
x=cos(2*pi*f1*n)+...
  cos(2*pi*f2*n);
X = fft(x)
```

FFT – digitalna frekvencija

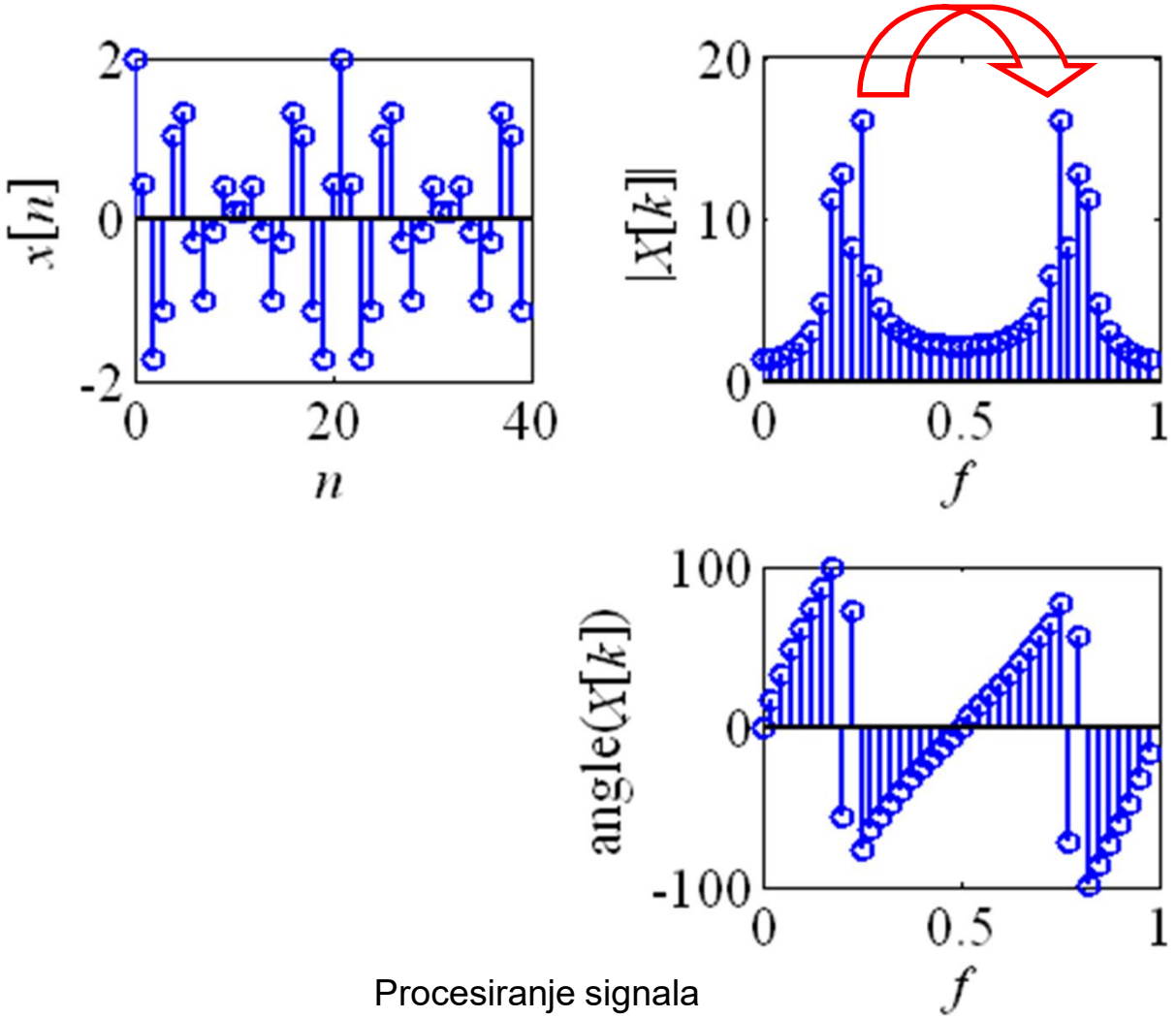




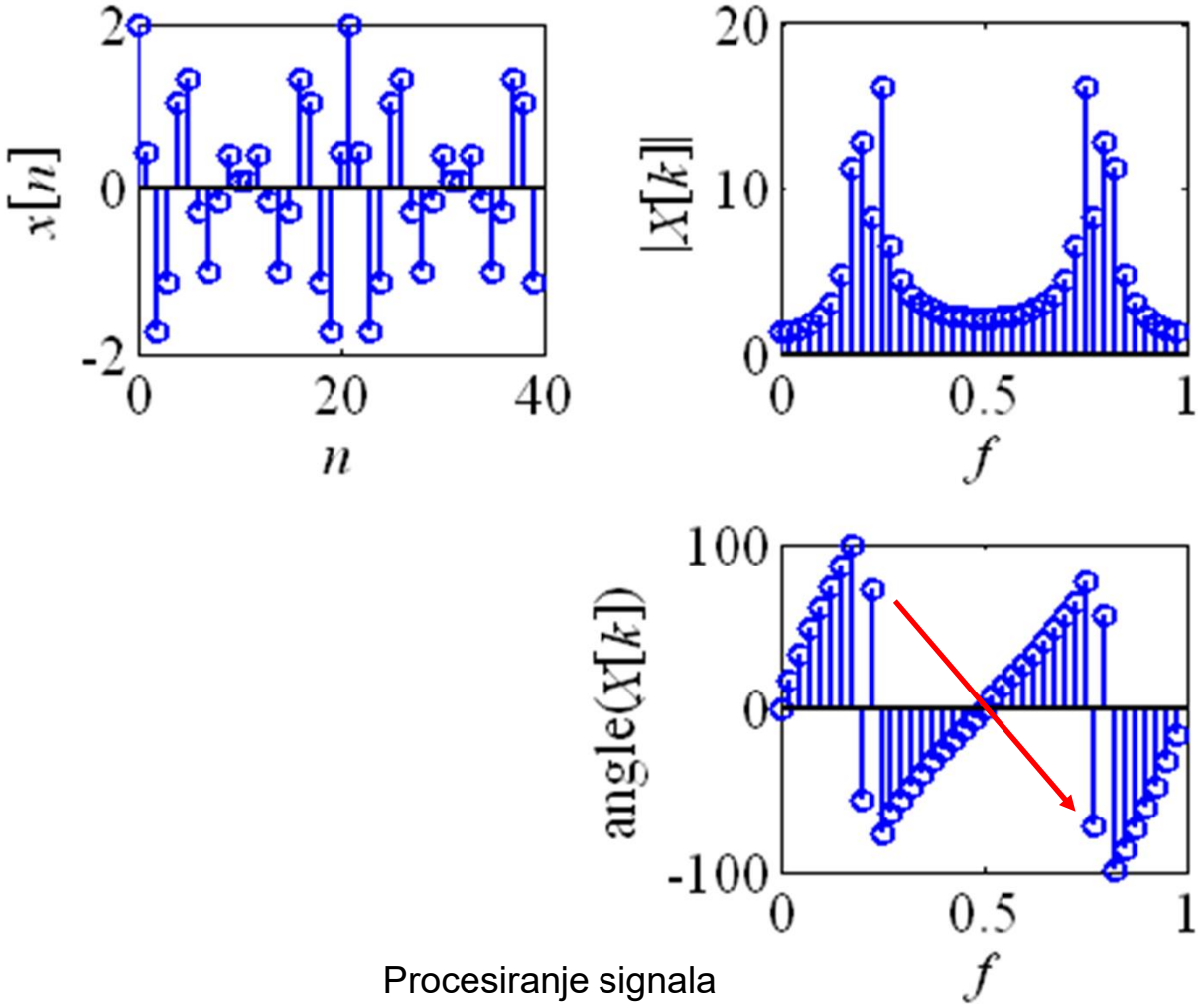

```
stem(n/N,abs(y))
xlabel('f')
```



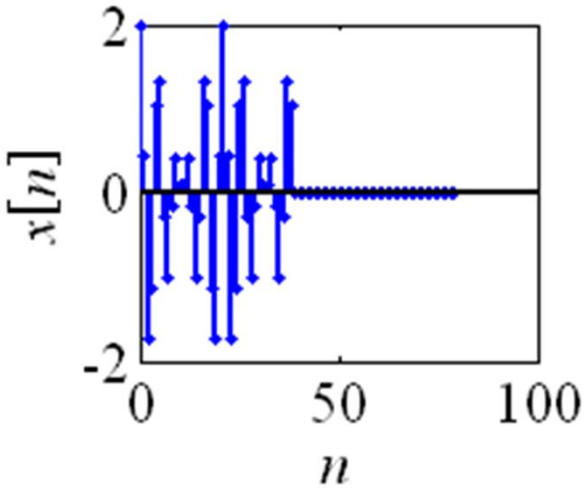
FFT – simetrija

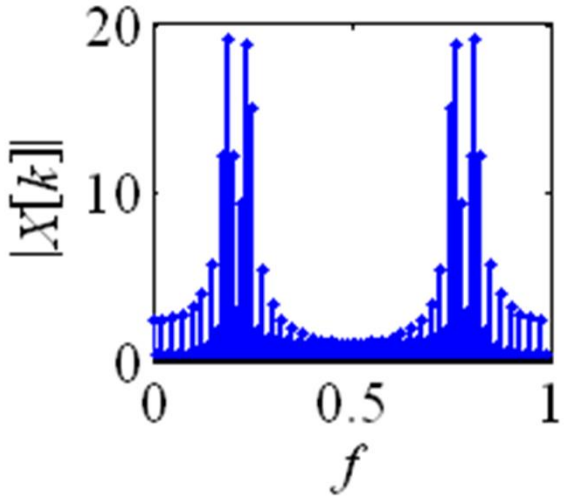


FFT – antisimetrija

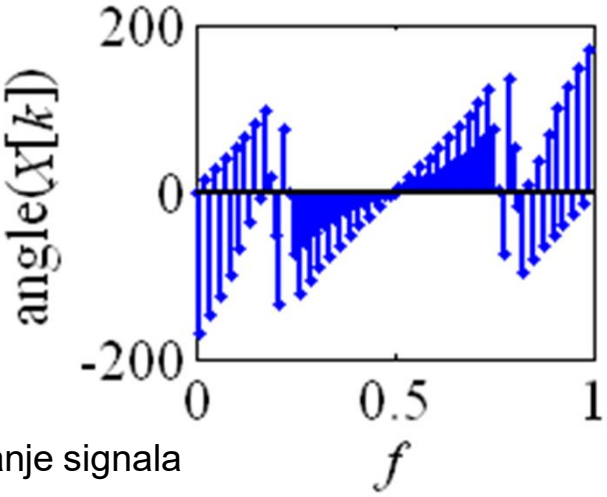


FFT – dopuna nulama

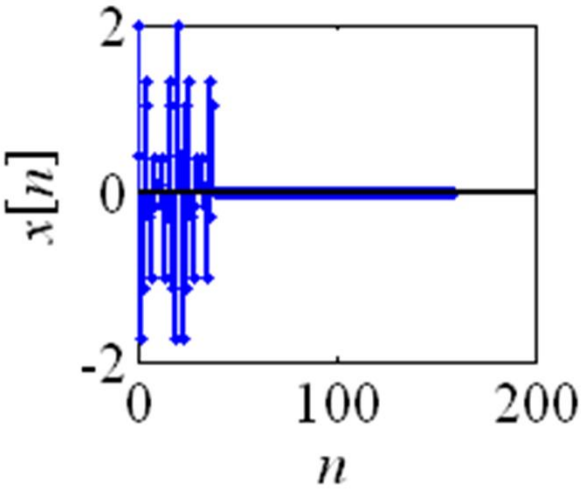


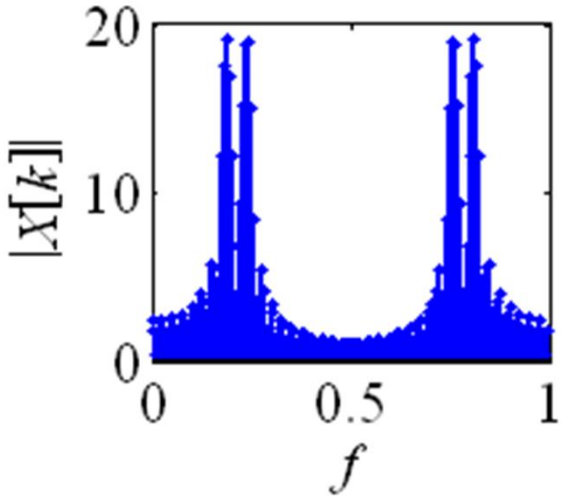



```
N = 80;
. . .
x = x.*(n<(N/2-1));
```

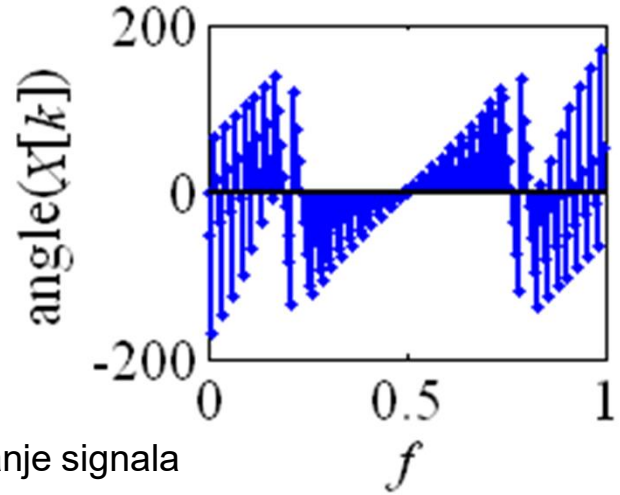


FFT – dopuna nulama (2)

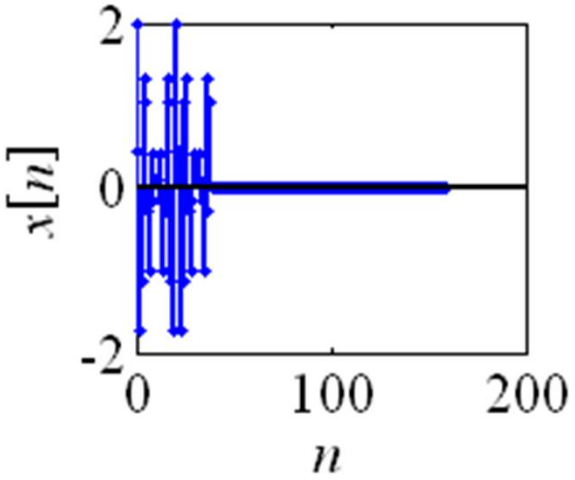


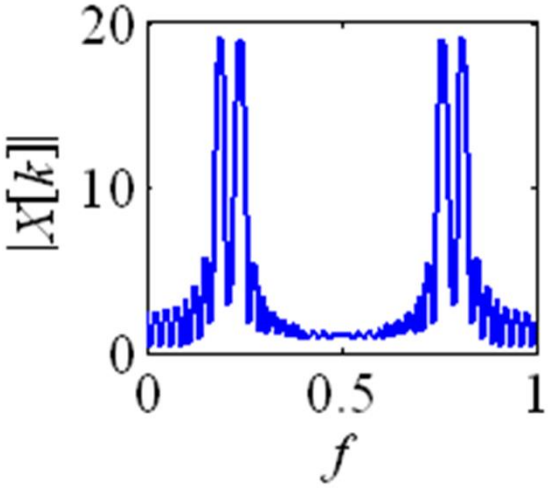


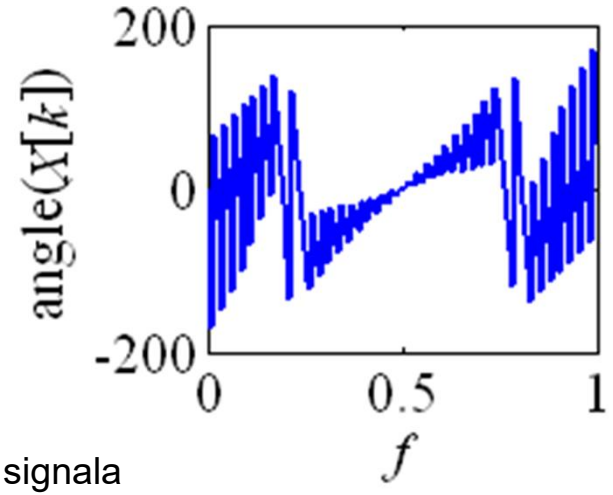

```
N = 160;
. . .
x = x.*(n<39);
```



FFT – dopuna nulama (2)

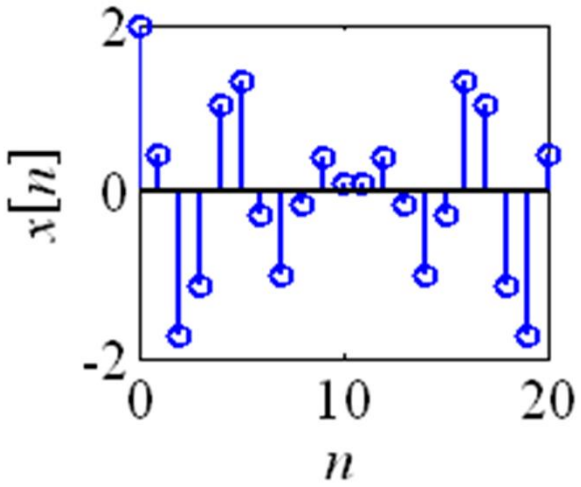


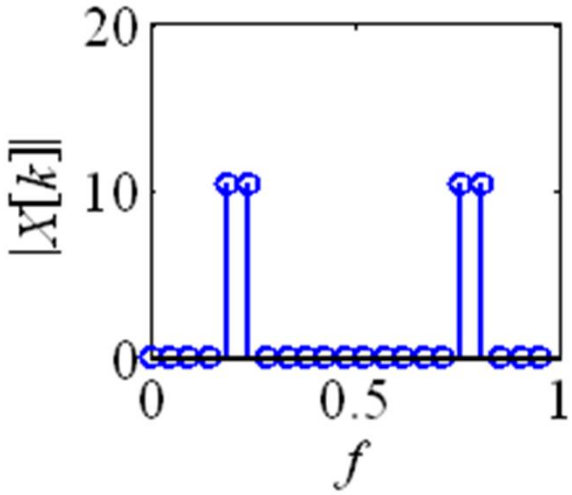




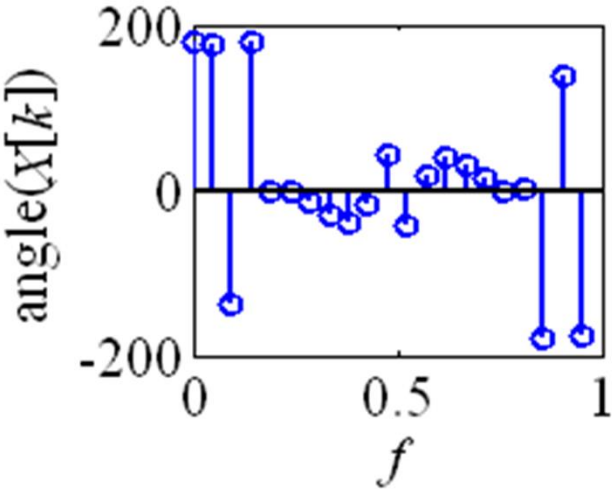

```
plot(n/N,abs(y),'-')
. . .
plot(n/N,angle(y)*180/pi,'-')
```

FFT – perfektna dopuna






```
N = 21;
. . .
k1 = 4;
f1 = k1/M;
k2 = 5;
f2 = k2/M;
x = cos(2*pi*f1*n)+cos(2*pi*f2*n);
x = x.*(n<21);
```



Parseval-ova teorema

Pojam snage

$$\sum_{n=0}^{N-1} |x[n]|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X[k]|^2$$

Profesor dr Miroslav Lutovac
mlutovac@viser.edu.rs