# Work # 1. Sampling theorem and aliasing

*Objective:* Acquiring of elementary skills of work with MATLAB package; examination of the sampling theorem and aliasing effect.

- **The theory basics**

The idea of the *sampling theorem* is the following: the continuous real signal with the spectrum limited by the frequency range $0 < f < f_m$ can be completely reconstructed from its discreet samples if the sampling frequency $f_s > 2f_m$.

The *aliasing* effect means that in case of sampling theorem violation (i.e. in case of too low sampling frequency), some frequency components of the signal cannot be differentiated from each other.

- **The main tasks of the work**

- To acquire elementary skills of work with MATLAB package in command and program modes (with the use of *m-files*). To learn how to calculate simple functions and represent them in graphical form.
- To examine the aliasing effect using artificially generated test signals and real electrocardiogram (ECG) signal.

- **The procedure of the work**

**Basics of work with MATLAB package.** For this and all the following works a special *work folder* should be organized to where all necessary signal files should be placed. The same folder should be used further for all program files and images that will be created in the course of each work.

1. Start MATLAB.
2. Set the path to your work folder.

MATLAB commands can be entered by two ways: immediately in the Command Window or with the use of text editor (see later). In case of *command mode,* all commands are executed immediately after they are entered, i.e.:

```
>> d=1+2 ↵
d =
     3
>> f=d+50 ↵
f =
    53
```

If a command is terminated by the character ";", the obtained numerical results (in case they are generated by this command) are not printed out in the Command Window.

3. Try to execute some simplest calculations similar to the shown above.

Command mode is applicable only in situations when not too much commands needed. To realize some complicated and voluminous calculation the *program mode* should be used. In this case, the necessary consequence of commands is prepared beforehand in the form of so-called m-file (text file containing all commands as separate lines). A special editor is used for the work with the m-files. This editor can be started by the New→Script button on the tollbar of the main MATLAB window.

A program prepared in the editor can be executed with the use of the Run button (green ► symbol at the toolbar). If some mistake is detected in the text of a program, the MATLAB types a message in the Command Window by red characters.

**Calculation and graphical representation of functions.** Below is an example of a program that illustrates how to calculate the function determined by the formula $y = \sin(3x)$ and how to draw a plot of this function.

```
X=0:0.02:2*pi;        % Preparing of the array (vector) of
                      % argument "X" values in the range from 0
                      % to 2π with the step 0.02 radians
Y=sin(3*X);           % Calculation of the sine function
figure(1)             % Window for the graphics output
plot(X,Y)             % Graphical output of the curve, with
                      % coordinates determined by the arrays
                      % "X" and "Y"
title ('Signal')      % The plot title
```

All text to the right of the symbol "%" is considered as *comments* and does not influence the program work.

1. Type the text of above program in the editor window (comments are not obligatory) and save the file with the extension "m".

2. Execute the program pushing the button ► at the tollbar.

3. Add the line for the second array calculation with the use of the following equation: $y = 2\cos(5x)$.

4. To plot this second function in the same graphic field use the following example:

```
hold on            % hold former curve at the graphical field
plot (X,Y2,'r')    % plot "Y2" array using red color ('r')
```

5. Create a title for the graphical window using Figure Name field of the panel that can be opened with the use of menu item Edit→Figure Properties.

6. Save the plot as "png" or "bmp" file using File→Save As item of the figure menu.
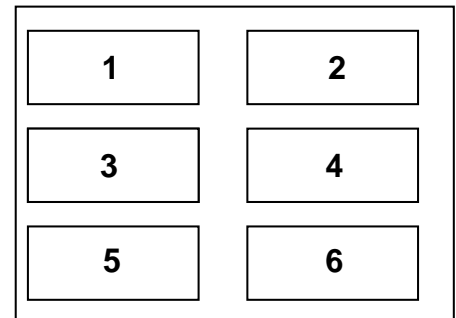
One graphical window can contain several fields for different plots. Below is the example of commands, creating graphical window including 6 fields (3 rows and 2 columns):

```
figure          % Create new graphical window
subplot(3,2,1) % Create 6 fields (3 rows and 2 columns)
```

This figure shows the positions and the numbers of the created fields.

To plot a curve in the field #4 the following commands can be used:

```
subplot(3,2,4) % Activate field #4
plot (Y)       % Plot "Y" array
```

```
┌─────────────────────────────┐
│  ┌──────┐      ┌──────┐      │
│  │  1   │      │  2   │      │
│  └──────┘      └──────┘      │
│  ┌──────┐      ┌──────┐      │
│  │  3   │      │  4   │      │
│  └──────┘      └──────┘      │
│  ┌──────┐      ┌──────┐      │
│  │  5   │      │  6   │      │
│  └──────┘      └──────┘      │
└─────────────────────────────┘
```

7. Add lines to your program that create new graphical window containing two fields in one row and plot the obtained above functions in these fields.

8. Create a title for the graphical window and save the plot as "png" or "bmp" file. Also, save the program text.

**Examination of the aliasing effect with the use of test signals.** Below is the text of program that plots two curves in one graphical field: sine function with the frequency $F = 30$ Hz and magnitude $A = 10$, sampled with the frequency $F_{s1} = 500$ Hz, and the same sine function, sampled with the frequency $F_{s1} = 100$ Hz. Time scales ($t_1$ and $t_2$) for the both these curves are from 0 s to 0.1 s.

```
F=30;                   % The sine function frequency (Hz)
A=10;                   % The sine function magnitude
tmax=0.1;               % Upper limit of the argument (s)
Fs1=500;                % The first sampling frequency (Hz)
Fs2=100;                % The second sampling frequency (Hz)
T1=1/Fs1;               % Sampling interval for the frequency "Fs1"
t1=(0:T1:tmax);         % Argument range for the first plot
T2=1/Fs2;               % Sampling interval for the frequency "Fs2"
t2=(0:T2:tmax);         % Argument range for the second plot

Y1=A*cos(2*pi*F*t1);% Sine function for the first plot
figure                  % Creation of the graphical window
plot(t1,Y1)             % The first plot output
hold on                 % Hold the first plot
Y2=A*cos(2*pi*F*t2);% Sine function for the second plot
plot(t2,Y2,'r')         % The second plot output
```

9. Create a new m-file and save it using some name.

10. Using the above program fragment as an example, create a program that calculates and plots in each of three graphical fields, placed in three rows, two curves of sine functions with the magnitude $A$ and the sampling frequencies $F_{s1}$ и $F_{s2}$ for the signal frequencies $F$, $F_{s2} + F$ and $F_{s2} - F$ (in upper, middle and lower fields correspondingly). The values of the parameters $F$, $A$, $t_{max}$, $F_{s1}$ and $F_{s2}$ for your variant are given in the table W.1.

If the last item is done correctly, the obtained plots show that when the lower sampling frequency ($F_{s2}$) is used, the values of sine signals in the points of samples are the same for all three cases. It illustrates the aliasing effect: three sine functions with obviously different frequencies cannot be distinguished, if the sampling frequency is too low.

11. Create a title for the window and save the plots as "png" or "bmp" file using the File->Save As item of the figure menu. Save the program text.

**Examination of the aliasing effect with the use of the ECG signal.** In the text files listed in table W.1 electrocardiogram (ECG) signal fragments are given. All these ECGs contains short pulses from the artificial pacing. The signals were recorded with the sampling frequency $F_S = 1200$ Hz.

12. Copy the ECG recording for your variant into your work folder.

The command loading ECG fragment from the file "W1_00.txt" is shown below:

```
Y=load('W1_00.txt'); % Read data from file to the array "Y"
```

The signals sampling frequency can be decreased with the use of the decimation procedure. The following example shows how the initial signal is decimated and plotted:

```
LY=length(Y);% Get the number of the read samples
Fs=1200;       % ECG sampling frequency
T=1/Fs;        % Sampling interval for the frequency "Fs"
tmax=LY*T;   % Signal fragment duration (s)
C1=6;          % Decimation coefficient
Fs1= Fs/C1;  % A lower frequency (here Fs1=1200/6=200 Hz)
T1=1/Fs1;    % Sampling interval for the frequency "Fs1"
t1=0:T1:tmax-T1;   % Time scale for the frequency "Fs1"
LY1=LY/C1;              % The number of samples for the frequency
                "Fs1"
for i=1:LY1             % Cycle for "i" from 1 to "LY1"
    Y1(i)=Y((i-1)*C1+1); % The decimated array samples
end
```

```
plot(t1,Y1)  % The decimated signal plot
```
13. Create a program that reads the needed file and plots in three graphical fields placed one below the other the following three signals: first with initial sampling rate 1200 Hz and two others that are decimated with the use of decimation coefficients $C_1$ and $C_2$ (take these values from the table W.1).

14. Expand the graphical window to all screen. If the obtained plots have different scales along the ordinate axis, set them to equal values using Edit→Axes Properties item of the graphical window menu. The same can be done immediately from the program, using the set operator, i.e.:

```
set(gca,'YLim',[-1.5 1.5]).
```

Here gca means reference to the current active plot field (get current axis), 'YLim' – is the plot property that determines the limits along the "Y" axis, and the values in brackets are the used limits.

If the last item is done correctly, the obtained plots show that the decimation of the signals samples (i.e. use of the lower sampling frequency) leads to the loss of the signal high frequency components (particularly the short pacing pulses) that represents a result of the sampling theorem violation.

15. Create a title for the window and save the plots as "png" or "bmp" file using the File->Save As item of the figure menu. Save the program text.

- **The report contents**

1. Title, goal and tasks of the work.
2. Texts of developed programs and all saved images.
3. Explanation of the results and conclusions.


- **Questions for the self-checking**

1. What is the main idea of the sampling theorem?
2. What does the aliasing effect represent and by what it is caused?
3. How the obtained plots of sine functions illustrate the aliasing effect?
4. Explain, using the given parameters of sine functions, in which certain cases the sampling theorem is violated and why?
5. Estimate the sine signals frequencies using only their plots.
6. Estimate the minimum value of the sampling frequency that is sufficient for the correct visualization of all sinusoids.
7. What is the decimation procedure idea and what does the decimation coefficient mean?

8. What conditions of the sampling theorem limit the possibility of the decimation procedure correct use?
9. Why are the short pacing pulses at the two lower ECG plots deformed or sometimes absent?
10. Estimate the minimum value of the sampling frequency that is sufficient for the reliable visualization of the pacemaker pulses?

- **Supplement #1**

Table W.1

| Variant | $F$, Hz | $A$ | $t_{max}$, s | $F_{s1}$, Hz | $F_{s2}$, Hz | File | $C_1$ | $C_2$ |
|---------|---------|-----|--------------|--------------|--------------|-----------|-------|-------|
| 1 | 13 | 2 | 0.5 | 200 | 45 | W1_01.txt | 3 | 10 |
| 2 | 15 | 5 | 0.3 | 300 | 50 | W1_02.txt | 2 | 15 |
| 3 | 24 | 20 | 0.2 | 350 | 80 | W1_03.txt | 4 | 25 |
| 4 | 26 | 25 | 0.2 | 400 | 90 | W1_04.txt | 5 | 16 |
| 5 | 28 | 3 | 0.15 | 450 | 95 | W1_05.txt | 6 | 20 |
| 6 | 30 | 10 | 0.1 | 500 | 100 | W1_06.txt | 2 | 25 |
| 7 | 35 | 50 | 0.1 | 600 | 110 | W1_07.txt | 3 | 16 |
| 8 | 35 | 6 | 0.1 | 800 | 130 | W1_08.txt | 5 | 20 |
| 9 | 40 | 10 | 0.1 | 900 | 150 | W1_09.txt | 4 | 15 |
| 10 | 45 | 7 | 0.1 | 1000 | 160 | W1_10.txt | 6 | 16 |