myFrequency
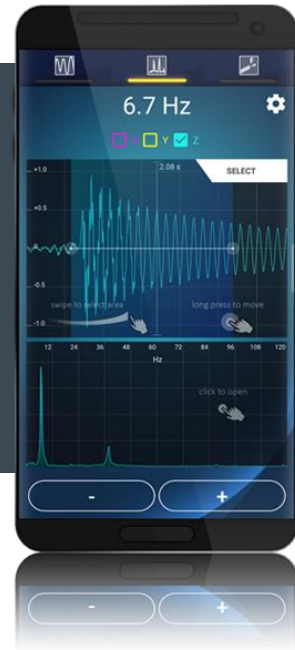
MATLAB

Brief instructions for importing the measurement files

APPTODATE | 2020

1   Introduction

First of all, we would like to thank you for choosing **my**Frequency – *The app for vibration measurements*.

With our MATLAB files, which are provided free of charge, we would like to offer you the option of subsequently opening and further processing your measurement files recorded with **my**Frequency on your PC.

In the following steps we would like to explain briefly and concisely how you import the measurement files into MATLAB and which functions we have already given you. Of course, you can expand the source code as you wish and add your own personal evaluation algorithms in order to adapt the measurement signals to your requirements and goals, process them and use them for your documentation.

Hint:

You need a valid license to use the MATLAB files, which you can purchase from the following website.

www.mathworks.com/products/matlab

Disclaimer of liability

APPtodate is not liable for the results determined with **my**Frequency, including the frequency spectra and time courses generated in MATLAB. If you as a user continue to use the determined data and represent it to third parties, you as a user are liable for the results.

APPtodate is not liable for property damage or personal injury that occurs in connection with the use of **my**Frequency. As the user, you are responsible for the correct assembly of the smartphone and must ensure that it is permanently installed so as not to endanger anyone. As a user, you do not have to expose yourself to any danger at any time and you are liable for any damage yourself.

APPtodate assumes no liability for possible data loss that can be traced back to the use of **my**Frequency or the MATLAB files provided.

By using our provided software, you agree to the points mentioned.

2	**Importing the measurement files**

After you have saved the desired measurement signals on your smartphone with ***my**Frequency*, you can transfer the measurement data to your PC either via email or via a USB cable.

Unzip the „myFrequencyAnalysis" – folder and save the unzipped folder under any path on your PC. After opening the folder you will see another subfolder with the name "Measuringdata" and five MATLAB files with the extension "-.m".

content of the folder *myFrequencyAnalysis*
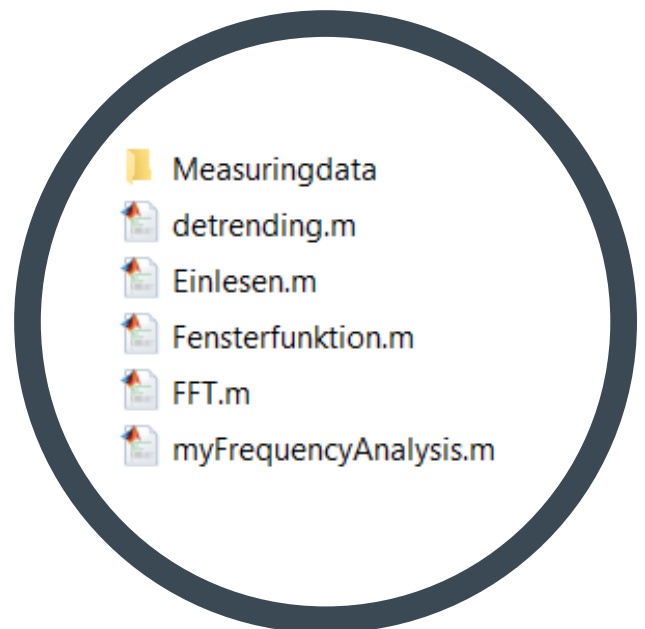
Copy the measurement files to be examined from the ***my**Frequency* app into the „Measuringdata" folder. **It is important that you do not rename the folder, as the MATLAB files in the program code have to access this folder and cannot find the measurement files if the "Measuringdata" folder is renamed or moved.**

Then run the MATLAB file "myFrequencyAnalysis.m". The MATLAB editor opens, in which you can write program code and, in this case, control the import of the measurement file. You have to adapt the lines of the source code shown on the following page in order to load the correct measurement file and set some parameters for the import.

The variable "name" is given the name of the desired measurement file that is to be imported. This measurement file here, for example "test.txt", must be located in the "Measuringdata" folder, as already explained. By changing this variable, you can address any measurement file in the "Measuringdata" folder.

source code – input of the user in the MATLAB editor

```matlab
% User-Input:
    % Here you have to write the name of the measurement you want to
    % analyze.
    name = 'test.txt';

    % Start- and Endvalue of the FFT-Calculation:
    % all = 1 : calculate the entire signal
    % all = 0 : calculate just a part of the signal
    all = 1;
    % if all = 0 : you can define a value for the beginning and the ending
    % (here you define the i-th value of the measurement)
    begining = 125000;  % this value is applied if you typed: all = 0;
    ending   = 127000;  % this value is applied if you typed: all = 0;
    % (Attention! : Be sure to set the begining and ending value in the
    % range of your measurment.

    % Set a 1 for Calculation, a 0 to avoid the calculation
    xAxis = 1;
    yAxis = 0;
    zAxis = 0;
    % Apply a window function to the signal to avoid capping effects:
    windowfunction = 0;
```

The following variable "all" can be assigned the value "0" and "1". If the variable takes on the value "1", the entire measurement file is loaded and a frequency spectrum is generated for the entire file. If the value "0" is assigned, a section of the measurement signal can be defined using the variables "begining" and "end". It should be noted that the i-th measured value of the signal is transferred to the variables "begining" and "end".

By assigning "1" and "0" to the variables "xAxis", "yAxis" and "zAxis" you can control which axes are to be imported and analyzed.

By assigning "1" and "0" to the "windowfunction" variable, you can control whether the measurement signal should be superimposed with a window function in order to avoid clipping effects at the beginning and end of the signal.

myFrequency

After assigning the named variables, the algorithm can be started using the "Run" button and MATLAB imports the selected measurement file and creates the desired time courses and frequency spectra of the selected axes.

button „Run" at the top of the screen

Example for selecting the start and end value:

At a sampling rate of 200 Hz (corresponds to 200 measured values per second) the 1000th measured value is exactly at 5 seconds.

> *begining   = 400;*
>
> *end   = 1600;*

In the present case, the algorithm starts at 2 seconds (400th value) and ends at 8 seconds (1600th value).

Attention:

When selecting the measuring range, make sure that the variables "beginning" and "end" are assigned values that are also contained in the measuring signal, otherwise an error will be caused.

3   Contact

If you have any questions or suggestions, please contact us directly, we will try to help you as soon as possible.

> *apptodate-center@hotmail.com*