Digital Signal Processing

Digital audio processing tutorial 2

Digital Audio Representation

By Yonghao Wang

Centre of Digital Media Technology



### **Learning Outcomes**

* + Understand the raw digital audio format in PCM codec.
  + be able to import windows WAVE (PCM format) file into Matlab raw data.
  + be able to analyse raw digital audio file in Matlab in terms of time domain parameters.

1. **WAV and PCM format**

The WAV or WAVE file format is a subset of Microsoft's RIFF specification for the storage of multimedia files. The file format is jointly designed by Microsoft and IBM. A RIFF file starts out with a file header followed by a sequence of data chunks. A WAVE file is often just a RIFF file with a single "WAVE" chunk which consists of two sub-chunks -- a "fmt" chunk specifying the data format and a "data" chunk containing the actual sample data which is shown in below figure:

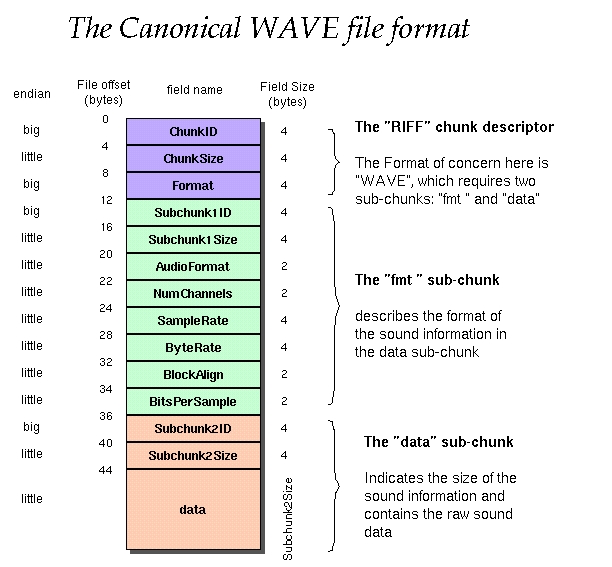


Figure 1 The WAVE format file structure

Although it is rare, WAVE file can be encoded in different format rather than Pulse Code Modulation (PCM) format, for example the loosy codec: Adaptive Differential (or Delta) pulse-code modulation (ADPCM). Therefore the value in “AudioFormat” field of the “fmt” sub-chunk indicates the actually codec. It is set to “1” if PCM is used.

PCM uses signed integer format to encode audio amplitude. The uncompressed PCM is also referred to Linear PCM. For example a 16bit PCM audio has maximum amplitude from -32768 to 32767.

Often software supports float point internally, the advantages are more accuracy and less processing errors using float point. Matlab natively uses double float point internally and the maximum value of audio will be converted into between -1 and 1.

1. **Import WAVE file into Matlab**

Matlab provides a built-in function called “**audioread( )**” to import audio file including the file with WAVE format into memory and split file information and raw audio data into different variables for further processing.

The official description for “**audioread()**” can be found in help file or online:

<https://uk.mathworks.com/help/matlab/ref/audioread.html>

This command is introduced after Matlab R2012b version. Please read carefully and understand all options of this function regarding WAV format. We will use it quite often in future tutorials. Especially, from the help file, you shall try to find out how to achieve these following two goals:

* How to read only part of the audio file?
* How to read audio file as “native” format?

1. **Tasks**

**Step 1 Download test wave file:**

The file “piano\_middle\_C.wav” can be downloaded from the module website at the current week course materials.

The time domain waveform of this audio looks like the following:

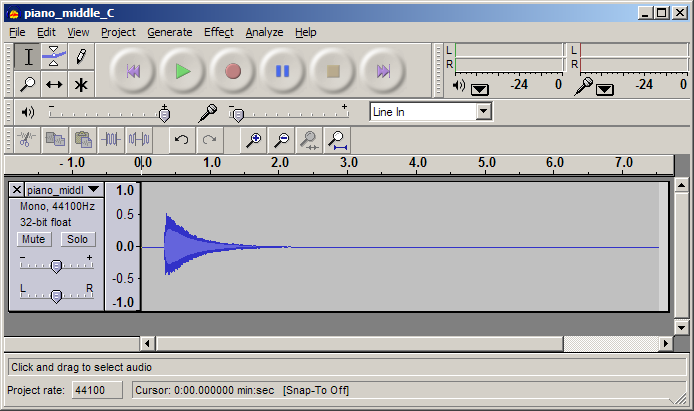


Figure 2 “piano\_middle\_C.wav” waveform view in Audacity

**Step 2 Code to import audio file**

In Matlab, typing the following code to import “piano\_middle\_C.wav” into some variables in memory. The words after % symbol are comments which are ignored by Matlab but they are useful for explanation of code.

%audio filename must in same directory of this m file

Filename = 'piano\_middle\_C.wav';

% First read wav file into memory

[Sig, Fs]=audioread(Filename);

%Sig stores raw audio data in column;

%Fs sampling frequency

Click “Run” to run this code and correct possible syntax errors.

**Step 3 Investigate and understand digital signal parameters**

In Matlab workspace investigate into those variables and fill the value and explain the physical meaning of them

* Fs - \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
* Sig - \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Calculate the duration of this signal using Matlab code, the function “length()” can be used to get the length of an array:

Duration = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

disp(Duration);

You can use “audioinfo()” command to get the information of the audio and compare with your observations.

**Step 4 Plot the Signal in time domain**

Complete the following code to plot this signal:

%audio filename must in same directory of this m file

Filename = 'piano\_middle\_C.wav';

% First read wav file into memory

[Sig, Fs]=audioread(Filename);

%Sig stores raw audio data in column;

%Fs sampling frequency

Duration = length(Sig)/Fs;

disp(Duration);

Ts = 1/Fs;

Time = 0:Ts:\_\_\_\_\_\_\_;

plot(\_\_\_\_, \_\_\_\_\_);

ylabel('Amplitude');

xlabel('Time Sec');

using command *sound()* correctly to playback the sound via Matlab code.

1. **Extended Tasks:**
2. In Matlab, calculate how many disk spaces or memory spaces this signal should occupy. Verify your answer with check the property or information of “wav” file in the computer.
3. Using following two methods to write Matlab code to get only part of original signal from 0.5 to 1 second length into Matlab workspace.
   1. Using *audioread( )* to input only from 0.5 second to 1 second.
   2. Using Matlab code to manipulate array of Sig and assign samples from 0.5 second to 1 second to a new variable “sig2”.
4. The “piano\_middle\_C.wav” is a mono audio file. Investigate the Matlab function of *audiowrite( )*, try to create a stereo audio file which contains left and right channels, each channel is same as this mono audio.