**Jim Allen**

**CS#&^ -- Small Computer Programming**

**Project #2 – WAV File Editor**

**WAVEditor v1.0**

**July 31, 2018**

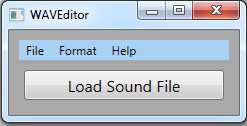
**WAV FILE EDITOR**

**WAVEditor v1.0**

**Overview**

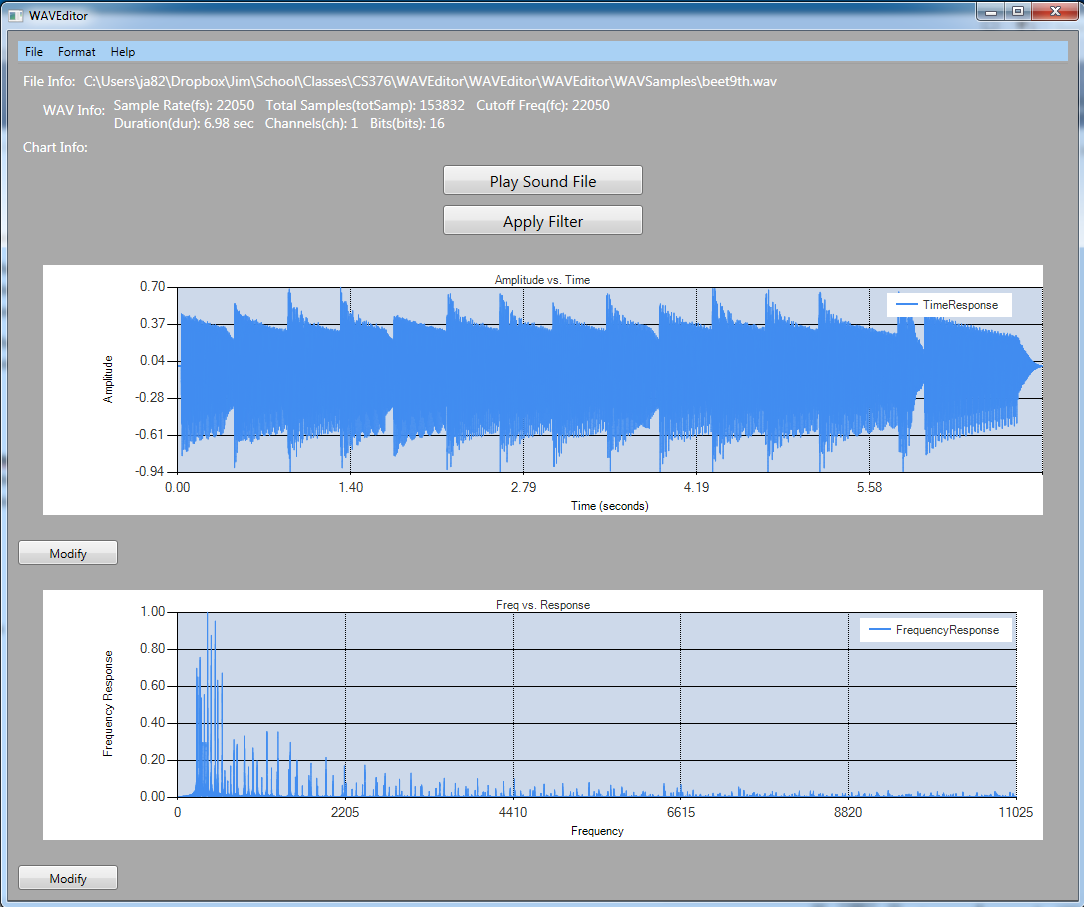
The WAV File Editor v1.0 was developed in July 2018 using Visual Studio 2013 running under Windows 7. MATLAB R2016a was used to provide audio file functionality to the C# application.

The purpose of this project was to develop a PC based application for parsing, playing, and filtering WAV files using intrinsic MATLAB R2016a functions.



**Figure 1 – WAVEditor Load Screen**

**WAVEditor v1.0** – the main program executable (Figure 1). See the Features section for more information about this application.



**Figure 2: Loaded WAV File**

**Features**

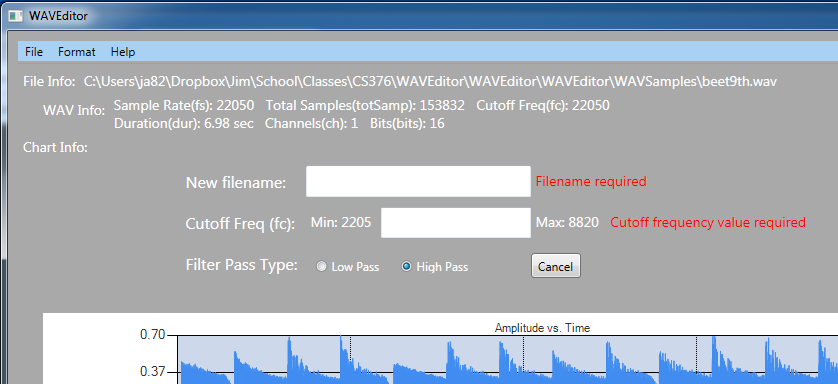
WAV File Info (See Figure 2):

The application allows for the convenient loading of WAV format files through a standard file selector dialog. Using the MATLAB functions MLWavInfo and MLReadWavFile, the header information is read from the file and then displayed in the summary box at the top of the window. Displayed information includes the sample rate (Fs), total samples, cutoff frequency, duration, channels, and bits.

Play Sound File Button (See Figure 2):

Upon successful loading of a WAV file, clicking the *“Play Sound File”* button calls the MATLAB function MLPlayWavFile which then plays the selected sound file.

Apply Filter (See Figure 3)



**Figure 3: Apply Filter Dialog**

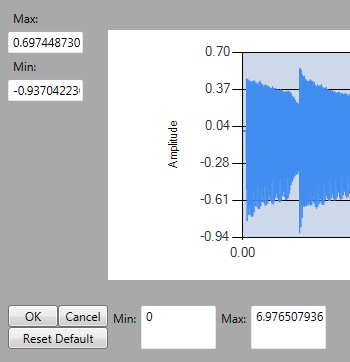
Clicking the *Apply Filter* button allows the user to create a new filtered file using either a low-pass or high-pass filter with a specified cutoff frequency. Dynamic error checking of input is used to ensure all required information is supplied with information regarding the source of the error. Upon successful completion of this mini-form, an “OK” button appears that then makes the changes to the original sound file and saves this to the specified filtered file name. This file is then immediately loaded and the plots are updated. Control is then returned to Figure 2.

Sound File Plots (See Figure 2)

***Time Domain Plot*** *–* Using the MATLAB function MLReadWavFile, a plot of time vs amplitude is made for the loaded sound file.

***Frequency vs Frequency Response Plot*** *–* Using the MATLAB function MLFreqResp, the file data is converted using Fourier transforms to create a plot of the frequency vs. frequency domain.

Modification of Plots (See Figure 4)



**Figure 4: Modify Plot Axes**

Each of the Time Domain and FTT plots contains a “Modify” button that enables a mini-dialog on the lower left of the plot that allows the user to change the upper and lower scale limits for the X and Y axis of the plots. The user may *CANCEL* the input by clicking the button, and apply the changes using the *OK* button. The user may also restore the default scaling with a single click of the *RESET DEFAULT* button.

Custom Menus (Figure 2)

***Format Menu*** *–* The format menu allows the user to change the color formatting of the foreground and background of the plots using a color selector dialog box.

***Help Menu*** *–* The Help menu provides basic information about the use of this program. In addition, an *About* features provides basic information about the creation of this application.

***Exit Menu Item*** *--* The *File* menu contains items to load a new sound file, and to exit the program.