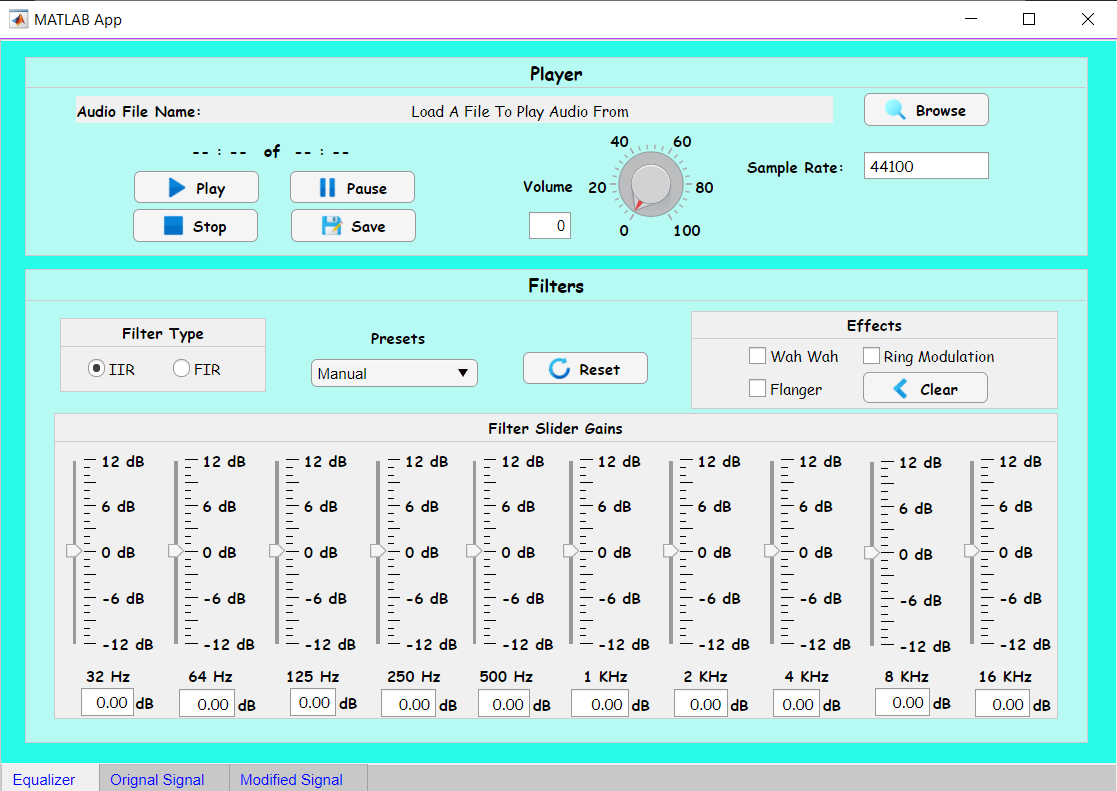
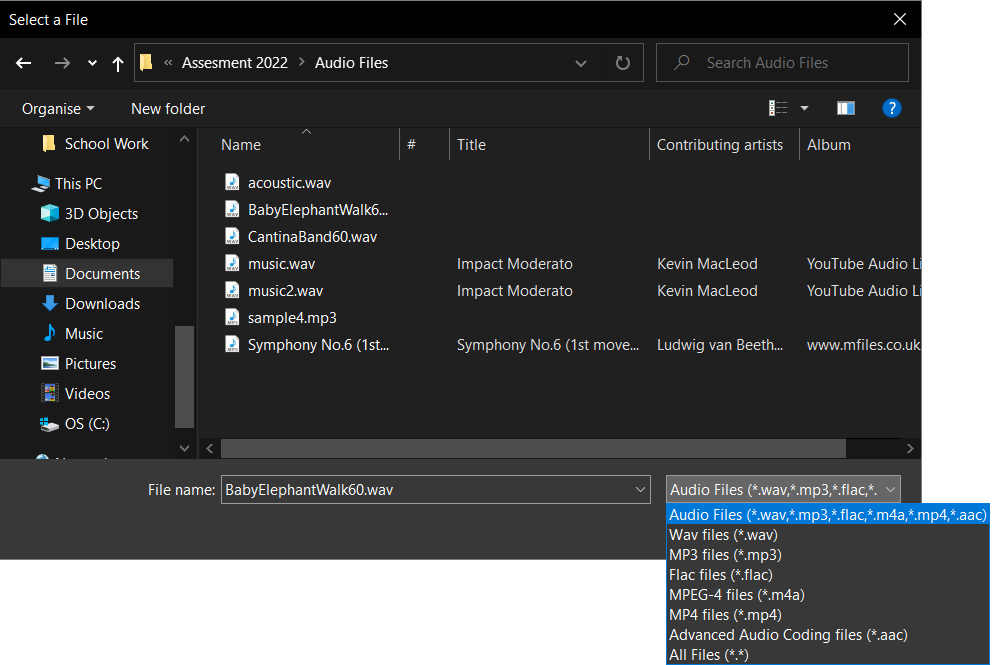
**MATLAB Based Graphic Equaliser**

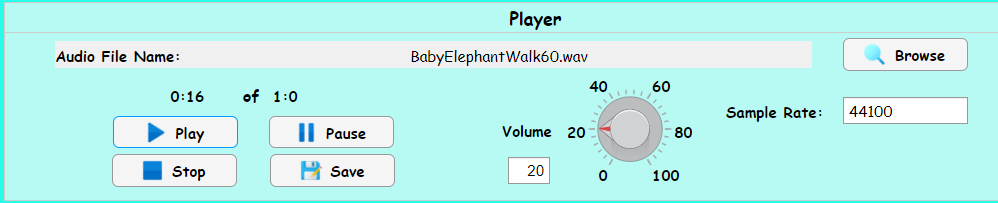
**Basic Overview of Design**

**Figure 1**

When the user runs the program, they are presented with a view as presented in **Figure 1**. The user is presented with many options in which the user first needs to load an audio file. Keeping the options flexible, the user can choose between a variety of audio file they are able to load in. For the volume a knob has been used which is similar to a volume knob in the real world. The user is able to change the sample rate at which the file is read by modifying the value in the sample rate field if not the default value of 44100Hz is used. The user is then able to play the audio file using the play button or the user can pause the audio file as well as completely stop the audio file from playing using the stop button. Giving the user flexibility the user can save the audio file after they have added filters to the original audio in which a new audio file is created and saved in the users current working directory. The user is able to choose between two different filter types IIR and FIR depending on the user’s needs. If the user is not concerned about linear phase characteristic, they can use IIR otherwise FIR would be the choice for linear-phase characteristics. Each filter type has its pros and cons but IIR is better for lower-order tapping, were as FIR is suitable for higher-order tapping. Making it easier for the user some common pre-sets have been preconfigured which will automatically adjust the sliders based on the pre-set chosen or the user can choose to manually adjust each slider. Saving the user time, a reset button has been added which will reset all the sliders back to zero. Lastly for the extra features the user can apply different effects to the audio by using the check boxes.

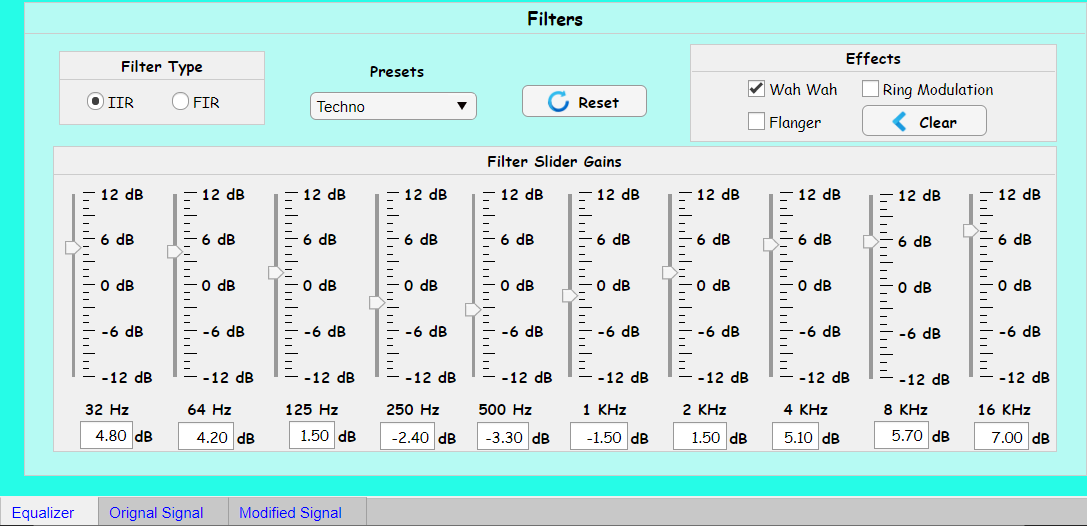
**Steps to get audio setup**

**Figure 2**

User first chooses a file as can be seen in **Figure 2**. Making the program more dynamic the user is able to choose between a variety of audio file types. File types present are: wav, flac, MP3 and MPEG-4 AAC.

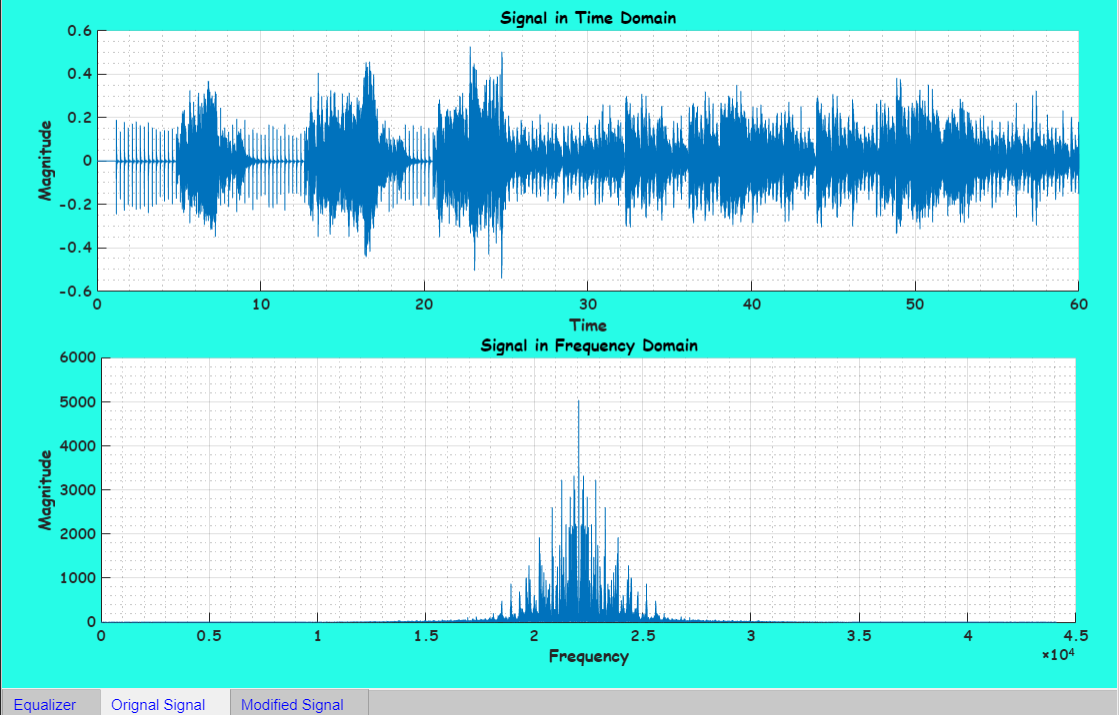
**Figure 3**

Once an audio file has been chosen the file name of the audio file is displayed to the user as can been seen in **Figure 3**. The user can the adjust the volume to their preference and also change the sample rate at which the file is being read. Once the user has adjusted the necessary setting the user can play the audio file. When the user plays the audio file a timer is displayed which shows the current playing time and the total time of the audio file.

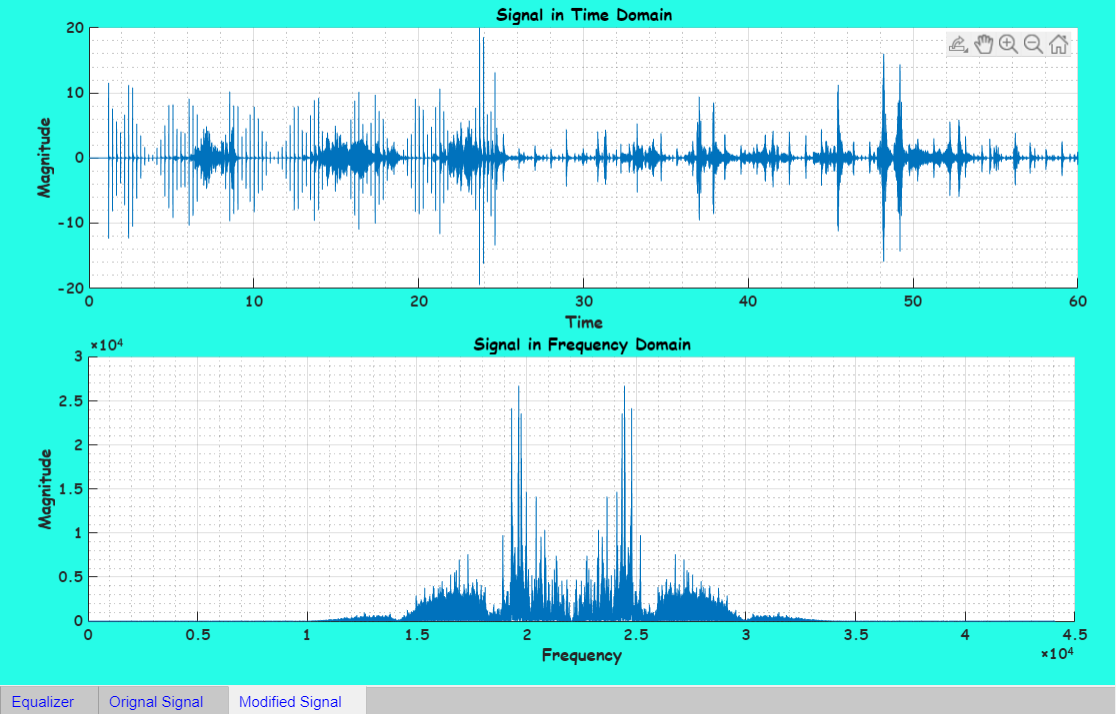


**Figure 4**

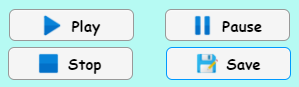
Once the user has loaded a file the user can choose a Filter Type, the user can select a pre-set and an audio effect to apply to the audio. The audio signal graphs of the original audio signal and the modified audio signal after applying different filters and effects can be view in the tabs that can be seen at the bottom in **Figure 4.**

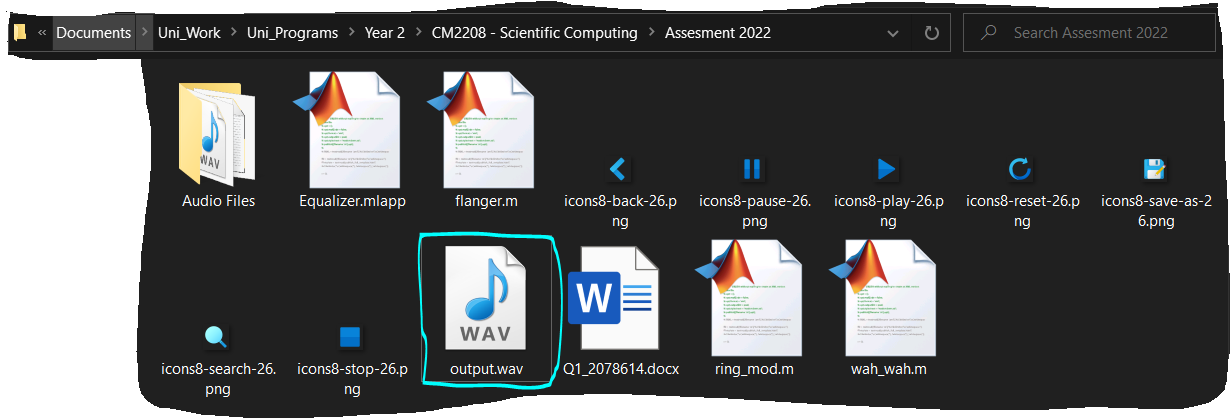
**Original Signal**

**Figure 5**

**Modified Signal**

**Figure 6**

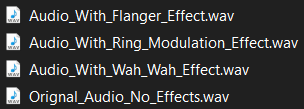
**Clicking the Save Button**

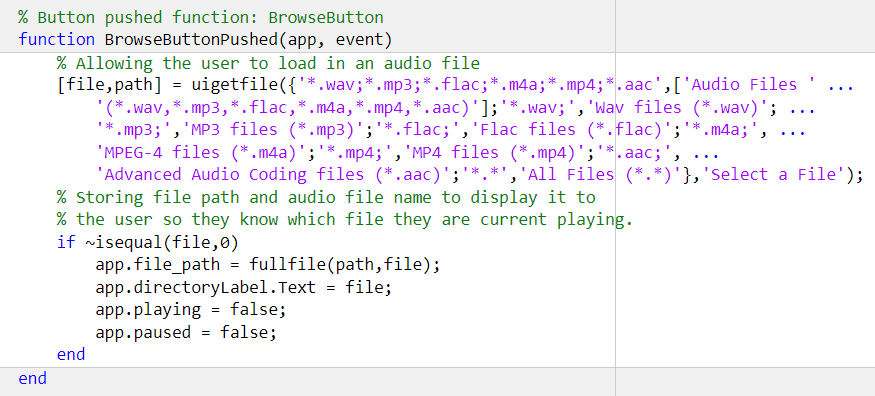




**Figure 7**

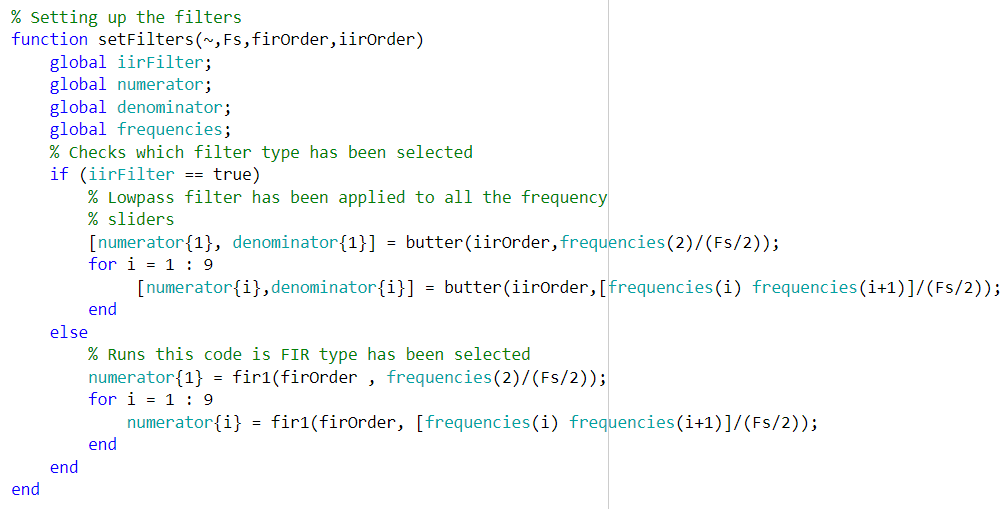
Once the user has applied effects to the audio file, they can save this new audio file. The new audio file is saved as “output.wav” (shown in **Figure 7**) in the current working directory which is opened in MATLAB.

Some of the saved files with different effects can be found in the folder shown.

**Basic Requirements Met**

**Figure 8**

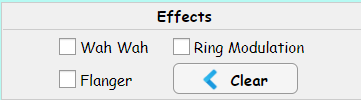
As shown before in **figure 2** and **figure 8** the user is able is able to select from different file formats and the system is able to read those different formats and play it back to the user.



**Figure 9**

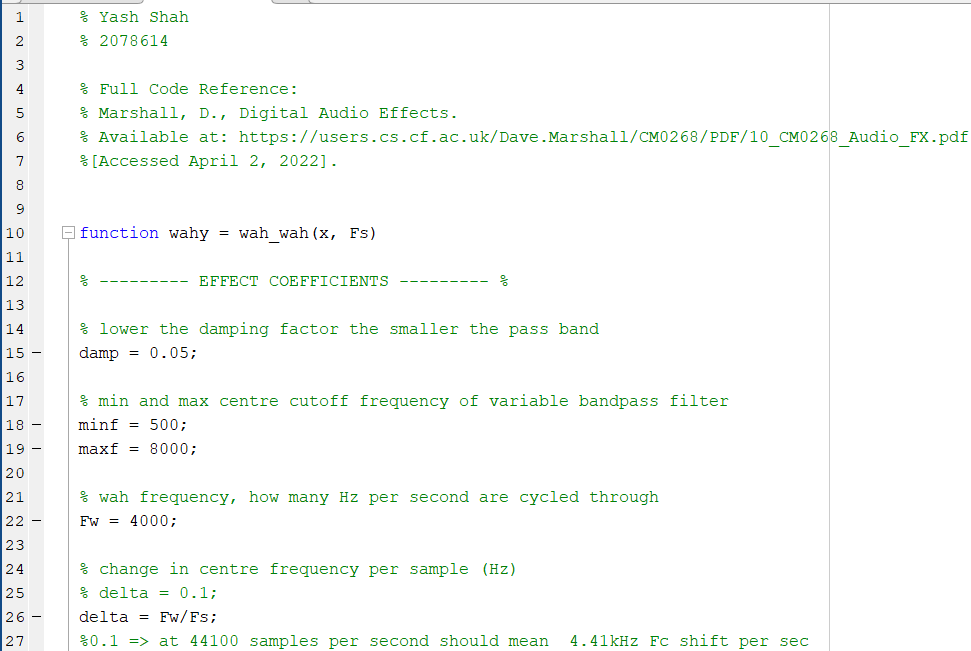
As shown in **figure 1** a multi-band graphic equaliser has been designed and the user is able to change the gain of each band frequency. As can be seen in **figure 9** the butter worth filter has been used which applies a low pass filter to all the bands if the user selects the IIR Filter Type. If the user selects FIR for the filter type, then a window-based FIR filter is applied to all the bands. In addition, the user is able to choose from preconfigured pre-sets for filtering the audio as shown in **Figure 1** and the user is also able to manually adjust the sliders for each band.

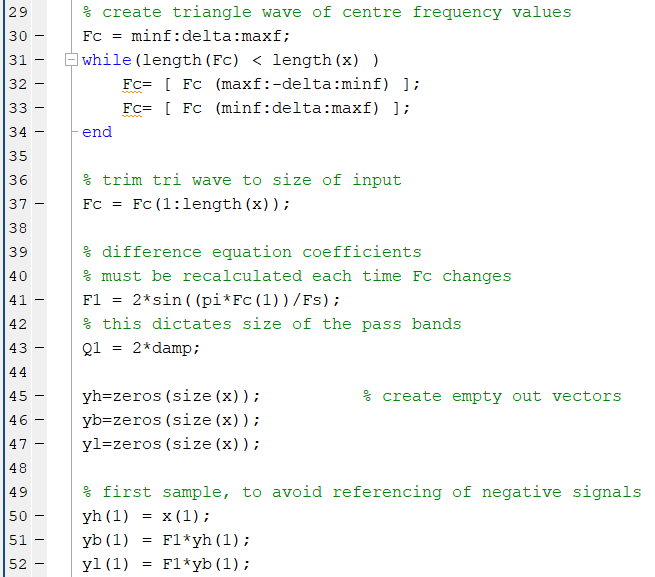
The user is able to adjust the volume of the audio play back using the volume control knob. The volume can only be adjusted between 0 and 100.

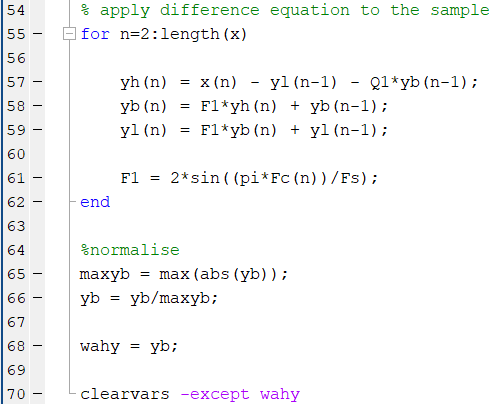
**Novel Extension**

For the novel features different effects have been implemented and the user can simply apply an effect by using the check boxes.

Code used for these different effects are found on Dave Marshall’s Digital Audio Effects slides.

**Wah Wah Effect**





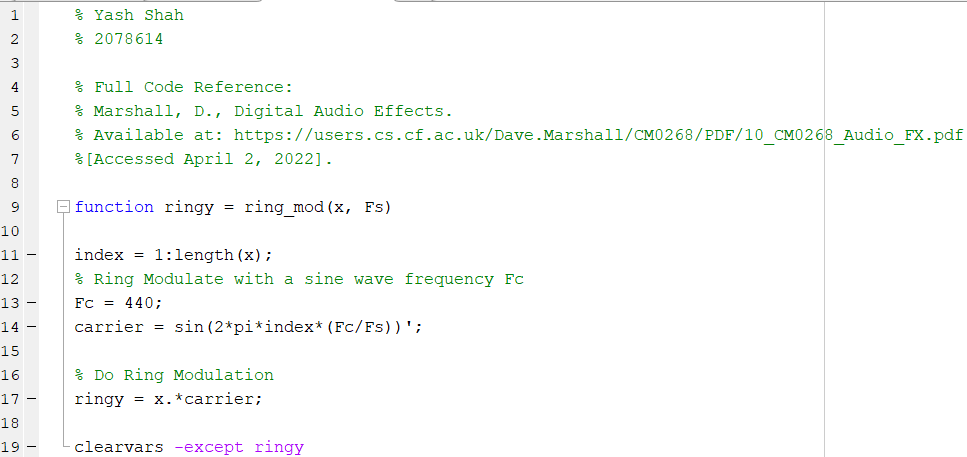
The code above performs three main functions:

1. Creates a triangle wave to modulate the centre frequency of the bandpass filter

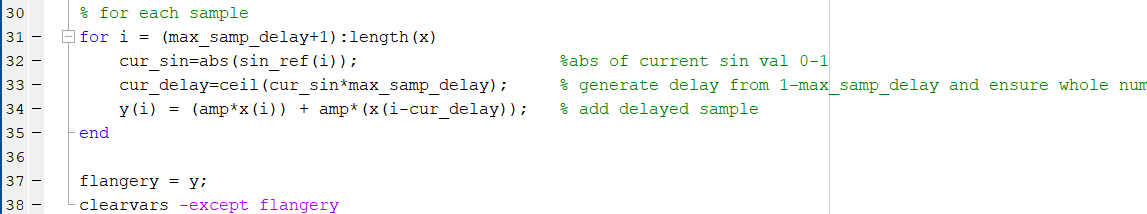
2. Implements a state variable filter.

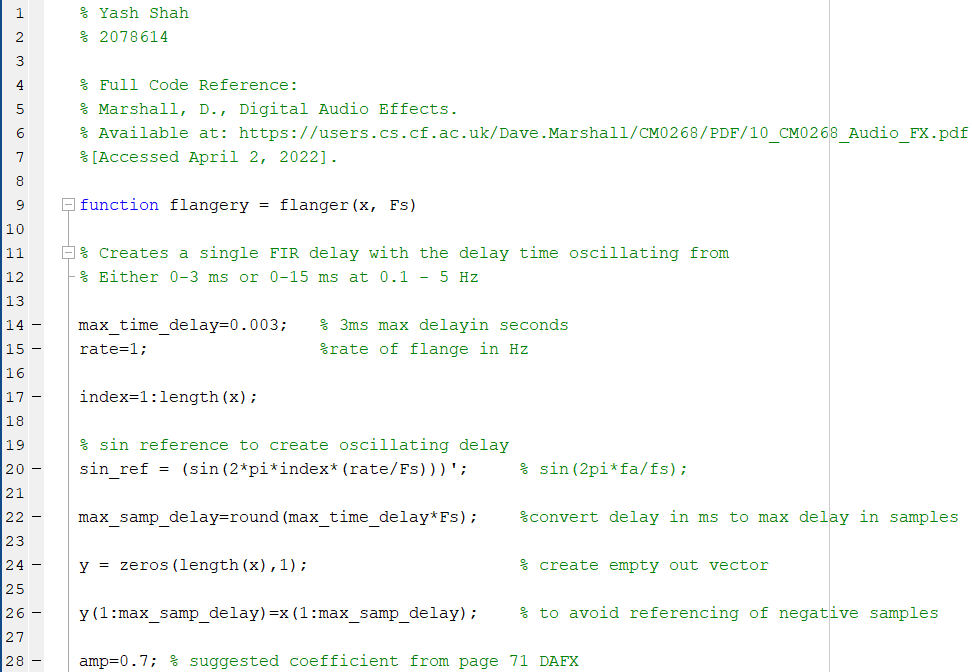
3. Repeated recalculation if centre frequency within the state variable filter loop.

**Ring Modulation**

****

The code above multiplies a sinewave with a carrier frequency FC with an audio modulator signal which produces an output wave. The sinewave is usually the carrier signal and the other signal which is the audio file is the input of the modulator signal.

**Flanger**

****

The code above works by mixing the same audio signal twice where once signal is played at a slightly slower speed. This creates the effect of two tape recordings playing simultaneously, but one going slightly slower than the other.

Reference

% Marshall, D., Digital Audio Effects.

% Available at: https://users.cs.cf.ac.uk/Dave.Marshall/CM0268/PDF/10\_CM0268\_Audio\_FX.pdf

% [Accessed April 2, 2022].