Using AudioMoth with Filtered and Triggered Recordings

theteam@openacousticdevices.info

11th May 2022

The most recent versions of the AudioMoth firmware (version 1.8.0) and configuration app (version 1.7.0) includes updated filtering and triggering options. In addition to the existing amplitude threshold trigger, a new frequency trigger is added that allows the response within a specific frequency band to be used to trigger recordings.

1 Filtered Recording

The AudioMoth firmware provides the option of applying first order Butterworth low-pass, band-pass and high-pass filters to acoustic data before it is written to the SD card. The type of filter and the cut-off frequency (or frequencies in the case of the band-pass filter) are set in the configuration app along with the other recording settings (see Figure 1).

The filters are applied in conjunction with a DC blocking filter which removes any small DC offset from the microphone readings and which is itself a Butterworth

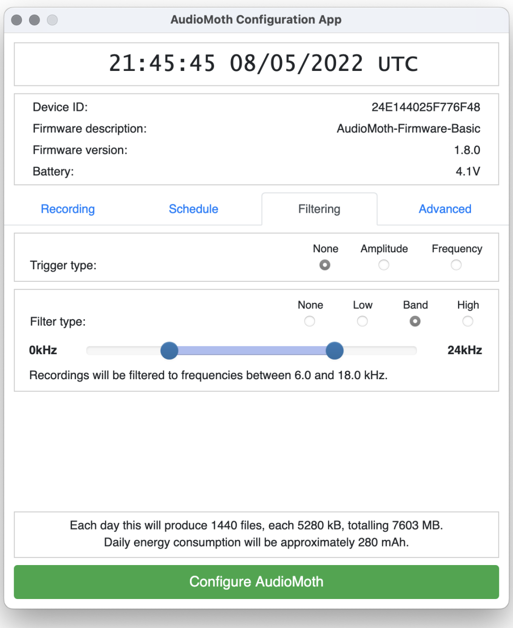

Figure 1: *Configuration App showing filter settings.*

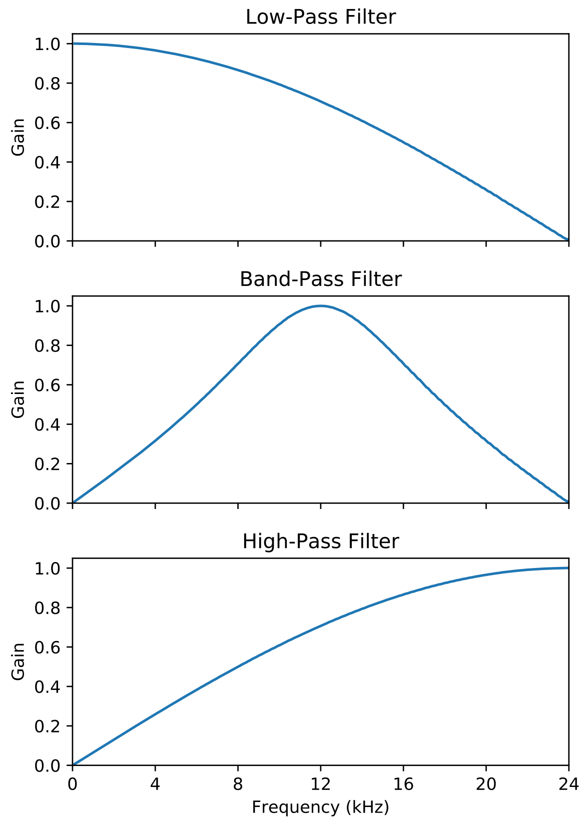

Figure 2: *Example 12 kHz low-pass, 8-16 kHz band-pass, and 12 kHz high-pass filters with 48 kHz sample rate.*

high-pass filter with a cut-off frequency of 48 Hz. The cut-off frequency of this filter can be reduced to 8 Hz using the *Disable 48Hz DC blocking filter* option in the configuration app. This allows very low frequency sounds such as elephant calls to be recorded. The filters are designed such that they exhibit close to unity gain at their peak frequency (see Figure 2).

The filters can be used to remove background noise and to improve the signal to noise ratio of recordings. For example, Figure 3 shows a typical band-pass filter which might be applied to improve the quality of bird song recordings. This is a band-pass filter with cut-off frequencies of 2 kHz and 12 kHz with a sample rate of 48 kHz. The lower cut-off frequency removes low-frequency noise, such as traffic, while the upper cut-off

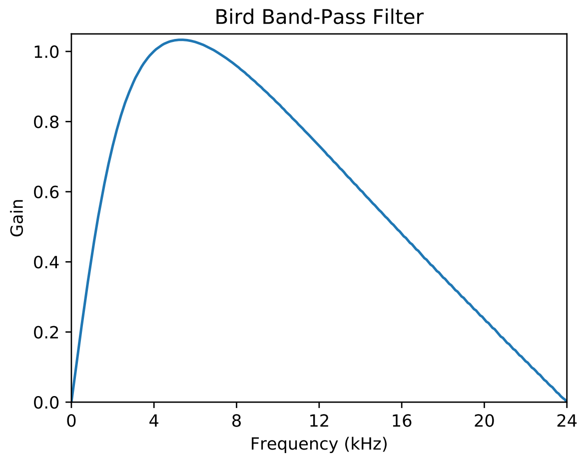

Figure 3: *Band-pass filter (2 kHz to 12 kHz) used for bird song recordings with a sample rate of 48 kHz.*

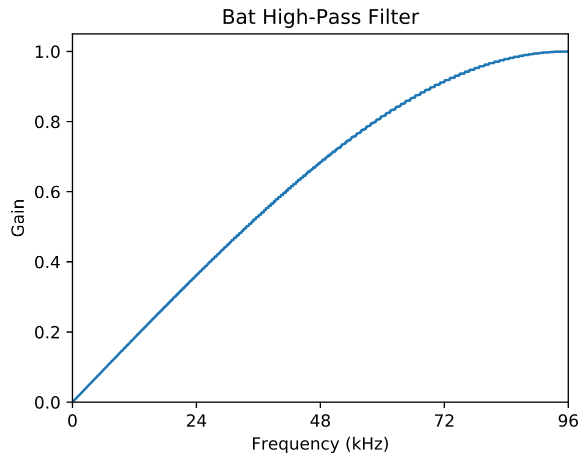

Figure 4: *High-pass filter (50 kHz) used for bat recordings with a sample rate of 192 kHz.*

frequency improves the signal to noise ratio of frequencies below 12 kHz (where most bird song occurs) by effectively averaging over neighbouring samples.

2 Triggered Recording

In order to save storage space, and also to reduce battery consumption, AudioMoth devices can make triggered recordings. Rather than making multiple individual files, which typically results in a loss of sample level timing accuracy, and blind periods whilst files are being opened and closed, AudioMoth writes triggered recordings to a custom T.WAV file format. The T.WAV file can be played and viewed with any WAV application. However, the position and length of all periods between triggering events are also stored in the file. This means that the silent unrecorded intervals can be re-inserted using tools within the configuration app to restore accurate sample-level inter-event timings. The single file can also be split into multiple shorter files as required for any particular processing task. See the *AudioMoth Triggered T.WAV File Format* application note for more technical details of this file format.[1]

The AudioMoth firmware makes the decision of whether or not to write each individual 32 KB audio buffer to the SD card. This decision can be determined by whether the amplitude of any sample within the buffer exceeds a threshold (amplitude threshold triggering) or whether the response within a particular frequency band exceeds a specified threshold (frequency triggering). Under both regimes, it is possible to specify a *minimum trigger duration* which determines the minimum duration of any event written to the SD card. The AudioMoth will continue to record for at least the specified duration after the last trigger event.

[1] `https://github.com/OpenAcousticDevices/Application-Notes/blob/master/AudioMoth_Triggered_T_WAV_File_Format/AudioMoth_Triggered_T_WAV_File_Format.pdf`

2.1 Amplitude threshold trigger

A typical use case for triggered recordings is to reduce the resulting file size when recording bats which make short passes over the deployed AudioMoth, leaving most of the recordings empty. In this case, we can use a filter to first reduce the background noise in the recording, and then apply an amplitude threshold. A 50 kHz high-pass filter works well for this. A band pass filter could also be used if the echolocation frequency of the target bat is known in advance. Figure 4 shows the characteristics of the 50 kHz high-pass filter and Figure 5 shows an example of its use. The background noise in this case is bird song and speech, and it is almost completely removed by the filter, leaving just the clearly distinguished bat calls. Since AudioMoth WAV files use 16-bit signed values, the possible values of samples are in the range from -32,768 to 32,767.

Having removed background sound, an amplitude threshold can be set to ensure that only those 32 KB segments which contain a sample that exceeds this value will be written to the SD card. The threshold can be specified as a 16-bit value, or as a percentage or dB value. The settings within the configuration app to do so are shown in Figure 6.

2.2 Frequency trigger

In addition to the amplitude threshold trigger, version 1.8.0 of the AudioMoth firmware introduces a frequency trigger. This approach uses a Goertzel filter to calculate a response within a particular frequency band, and allows a specified threshold to determine whether or not each 32KB audio buffer should be written to the SD card when generating the T.WAV file. As described below, the frequency band is defined by a centre frequency and a window length (measured in samples) that defines its width.

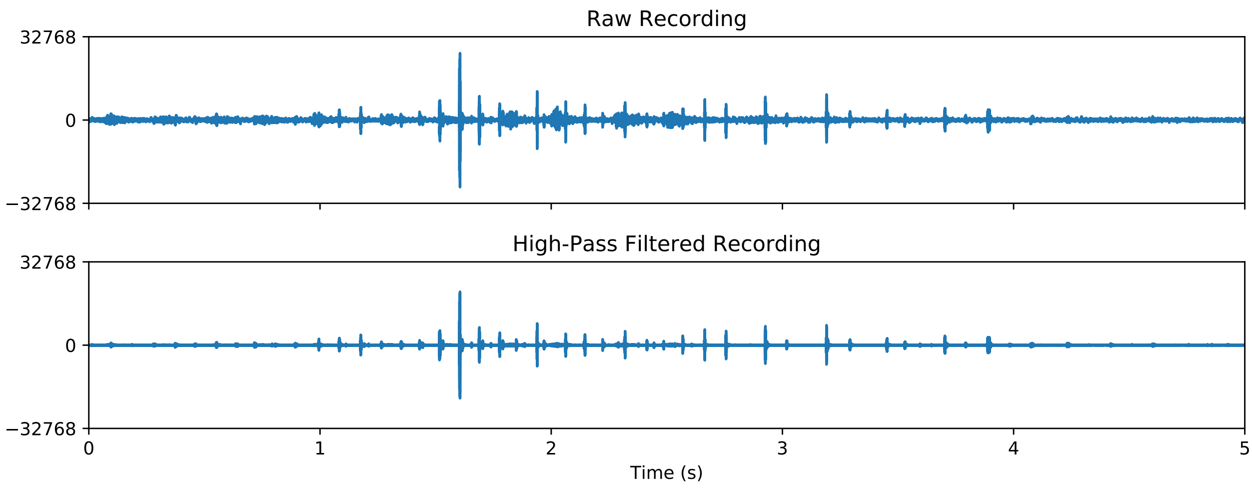

Figure 5: *Example of raw and 50 kHz high-pass filtered recordings showing the reduction in background noise.*

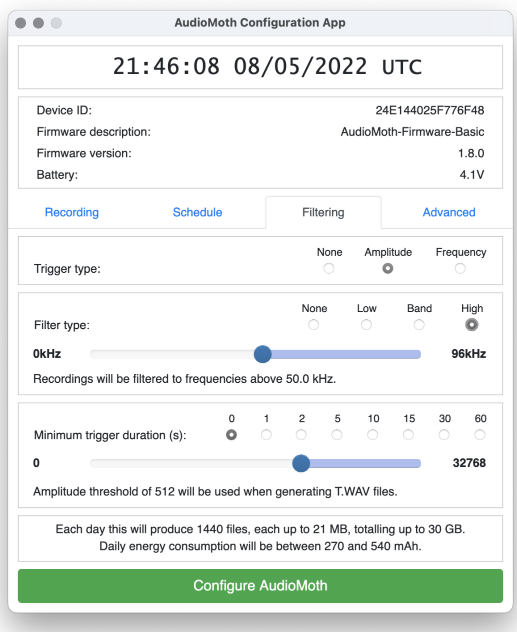

Figure 6: *Configuration App showing amplitude threshold triggering settings.*

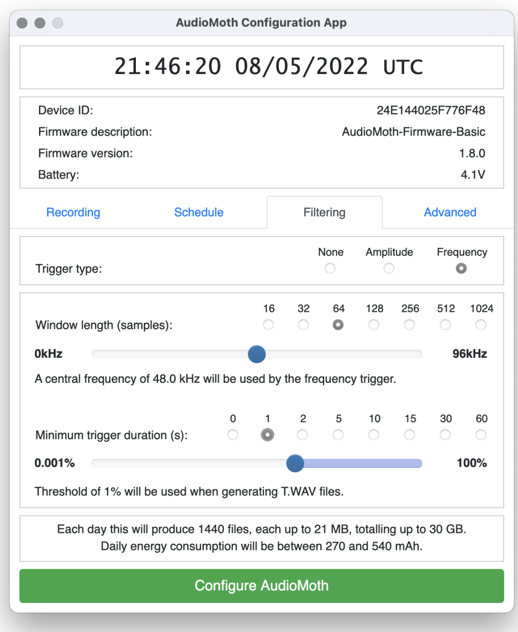

Figure 7: *Configuration App showing frequency triggering settings.*

2.2.1 Goertzel filter

A Goertzel filter efficiently calculates a single term of a discrete Fourier transform (DFT). To do so, it operates on a window of N samples, and calculates an iterative term given by:

$$y_n = (h_n.s_n) + (c.y_{n-1}) + y_{n-2} \qquad (1)$$

where $0 \leq n < N$ and s_n is the microphone sample, normalised to a scale of -1 to 1. The value of c is given by:

$$c = 2 \times \cos\left(\frac{2\pi f}{f_s}\right) \qquad (2)$$

where f is the centre frequency of the band and f_s is the sample rate. Both y_{-1} and y_{-2} are initialised to zero. The Hamming window coefficients are given by:

$$h_n = 0.54 - 0.46 \times \cos\left(\frac{2\pi n}{N-1}\right) \qquad (3)$$

Once N samples have been evaluated, the magnitude of the signal within the frequency band is given by:

$$m = \sqrt{y_{N-1}^2 + y_{N-2}^2 - c.y_{N-1}.y_{N-2}} \qquad (4)$$

This magnitude can be multiplied by the factor:

$$\frac{2}{\sum_{n=0}^{N-1} h_n} \qquad (5)$$

to yield a maximum response of 1 when evaluated on a full-scale sine wave of frequency f.

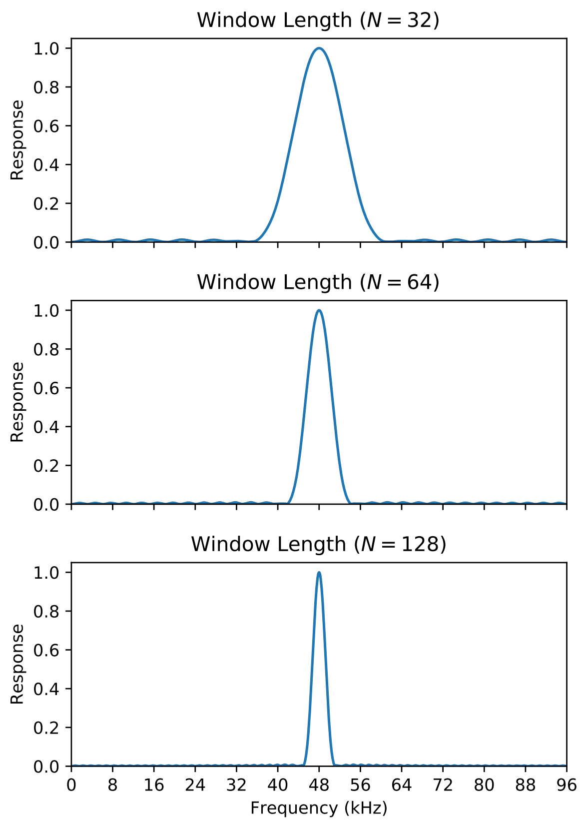

Figure 8: *Example frequency bands with window lengths of 32, 64 and 128 samples at 192 kHz sampling.*

The width of the frequency band is determined by the bandwidth, B, of the Goertzel filter, and is given by:

$$B = 4\frac{f_s}{N} \tag{6}$$

Note that as N increases, the width of the frequency band decreases. Figure 8 shows the response within three frequency bands using window lengths of 32, 64 and 128 samples when a full-scale sine wave of varying frequency is applied to a Goertzel filter with a centre frequency of 48 kHz. Figure 7 shows the corresponding frequency trigger settings within the configuration app.

3 Expanding T.WAV Files

The T.WAV files generated by amplitude threshold or frequency triggering can be expanded using tools within the configuration app (see Figure 9). These tools insert silent periods, of the correct length, between triggered events in order to restore the correct sample-level timing of the equivalent non-triggered recording. Settings within the configuration app allow additional options, such as generating multiple files with a given maximum size, which may be useful depending on the ultimate use of the recordings.

The expansion tools can be used in many different ways. One option is to record multiple short files and then use the size of the resulting recordings as an indicator of their content. For example, Figure 10 shows

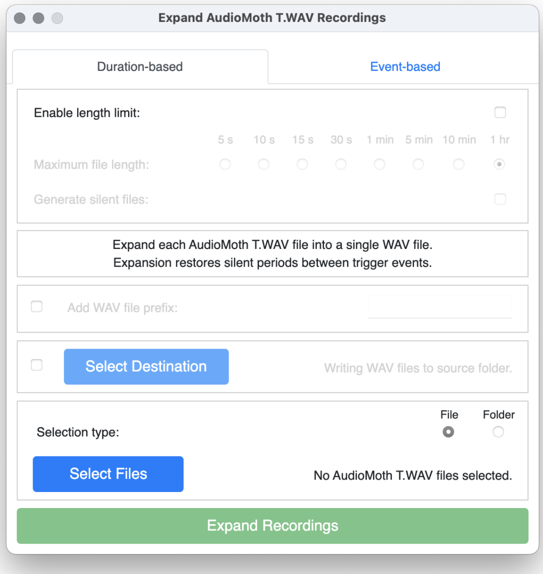

Figure 9: *Configuration App showing T.WAV file expansion window.*

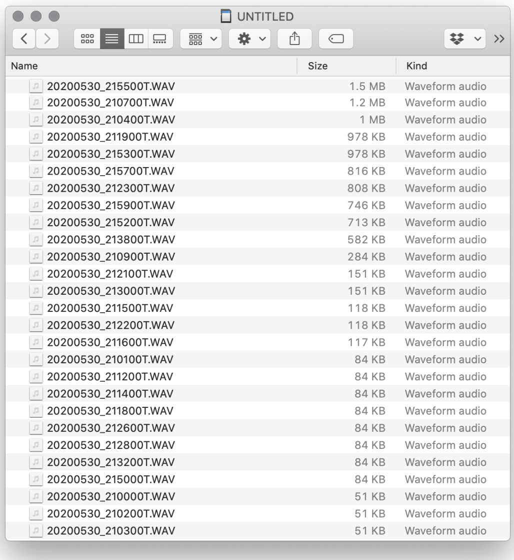

Figure 10: *File sizes during 1 hour of data collection using a 55 second record and 5 second sleep cycle.*

the result of using a 55 second record and 5 second sleep cycle for one hour with a sampling rate of 192 kHz, a 50 kHz high-pass filter and a threshold of 512. These recordings would each normally be 21 MB in size, resulting in over 1.2 GB of data in one hour. However, in this case, the total size of all 60 files is just 13 MB, reducing the total file size by a factor of close to 100, and enabling a quick scan of the file sizes to indicate periods of bat activity.

Figures 12, 13 and 14 show an example of the results of expanding a T.WAV file. Figure 12 shows the original `20200530_210400T.WAV` file recorded by the AudioMoth. This file is 1 MB in size and on examination

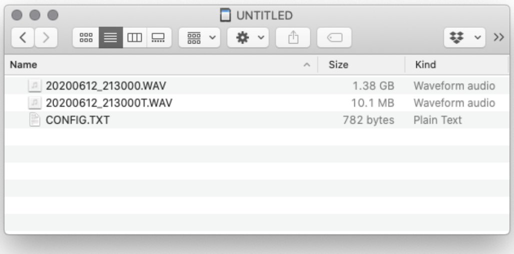

Figure 11: *File sizes during 1 hour of continuous recording.*

contains just over 2 seconds of audio. The sonogram clearly shows a bat passing over the AudioMoth with a call around 50 kHz.

Figure 13 shows the resulting `20200530_210400.WAV` produced by the default expansion settings. This file is 21 MB in size and has been expanded out to the original 55 seconds. The sonogram shows a single bat pass that occurred 35 seconds into the recording.

Figure 14 shows the same expanded `20200530_210400.WAV` file, but this time zoomed in to the period around the 35 second marker. Over this period the correct timings of the individual echo location calls are restored. In this case, each 32 KB segment is approximately 85 milliseconds in length (calculated from 32 $\times$ 1024 bytes / 2 bytes per sample / 192,000 samples per second = 85 milliseconds), and there are several segments within the call, as well as those on either side of it, which did not contain a sample whose value exceeded the threshold.

An alternative approach is to make one continuous triggered recording over the period of interest. Figure 11 shows the results of recording a one hour period, again with a sampling rate of 192 kHz, a 50 kHz high-pass filter and a threshold of 512. The captured file is just 10 MB in size and contains approximately 26 seconds of audio (see Figure 15). This can be expanded to the full one hour recording resulting in a file which is approximately 1.4 GB in size (see Figure 16). This large file can be viewed and trimmed as necessary, for example, around the bat pass which occurs 42 minutes into the recording (see Figure 17). Alternatively, further options in the configuration app allow the expanded file to be split into multiple files with a specified maximum length.

Finally, Figures 18 and 19 show an expanded 55 second frequency triggered recording which uses a minimum trigger duration of 1 second, a centre frequency of 48 kHz, a window length of 64 samples and a 1% threshold value. In this case, the minimum duration of any recorded period within the expanded file is 1 second, and the quieter feeding buzz that follows the bat call in Figure 19 is recorded when it may not have triggered a recording itself. Note also, that this recording show more low-frequency noise than the previous recordings, due to the absence of the high-pass filter.

4 AudioMoth Filter Playground

Choosing filter, amplitude threshold and frequency triggering settings requires some care. It can often be performed on a trial basis, and to assist this process, the AudioMoth red LED will only flash when the device is actually writing data to the SD card. This can be used to judge whether or not background noise is triggering the recording. With both the amplitude threshold and frequency triggering, increasing the threshold value will reduce triggering from background noise.

A more direct approach is to collect standard non-triggered recordings of the environment under study and then then use the AudioMoth Filter Playground website (`https://playground.openacousticdevices.info/`) to explore the effect of filter and triggering parameters (see Figure 20). The website exposes all the same controls as the configuration app and the displayed waveform and sonogram responds instantly to changed settings. The final configuration can then be exported from the filter playground to the configuration app to aid in setting up AudioMoth devices for a later deployment.

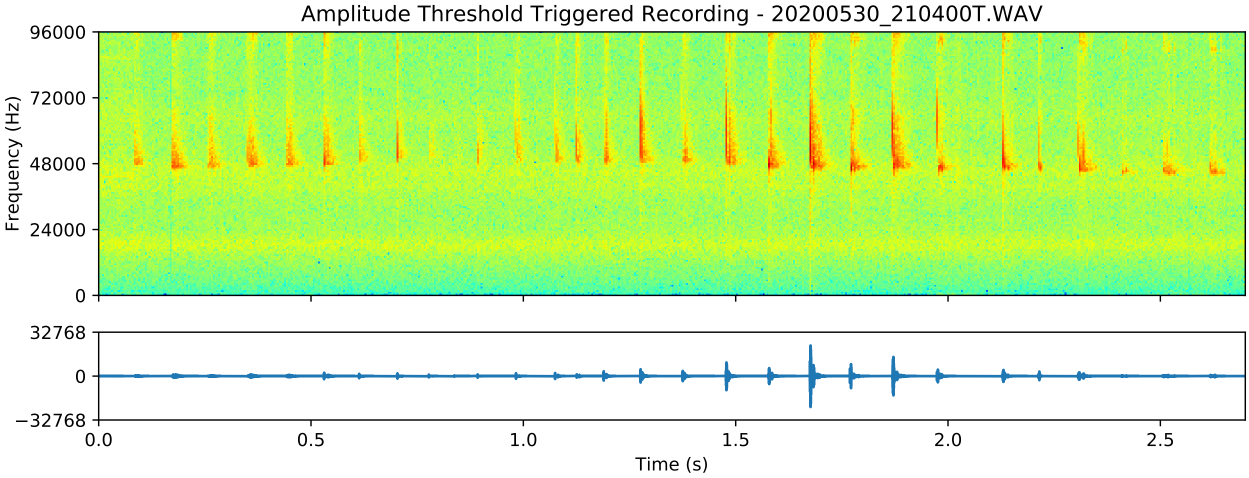

Figure 12: *Original* 20200530_210400T.WAV *file.*

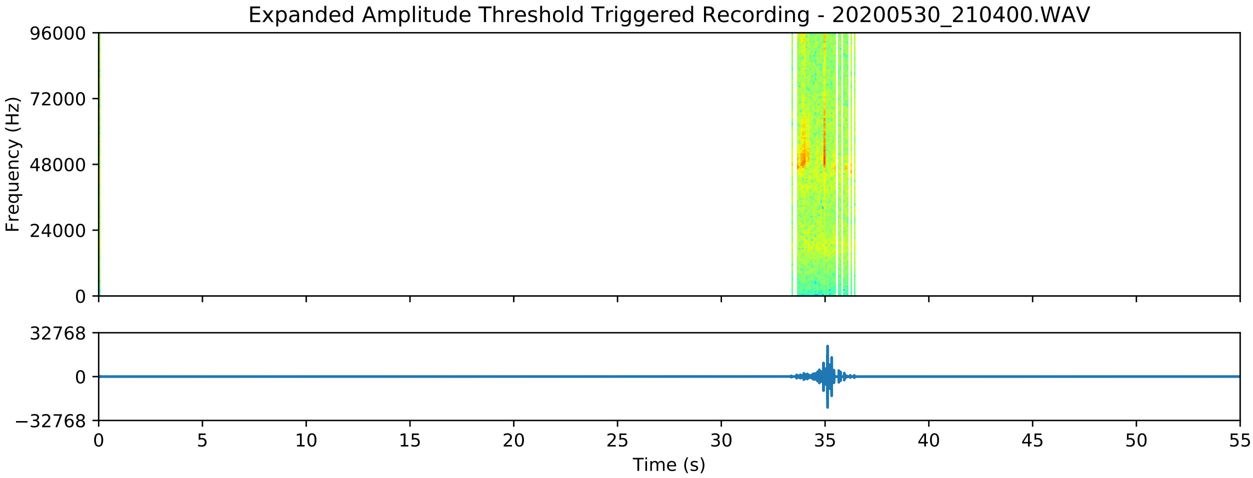

Figure 13: *Expanded* 20200530_210400.WAV *file.*

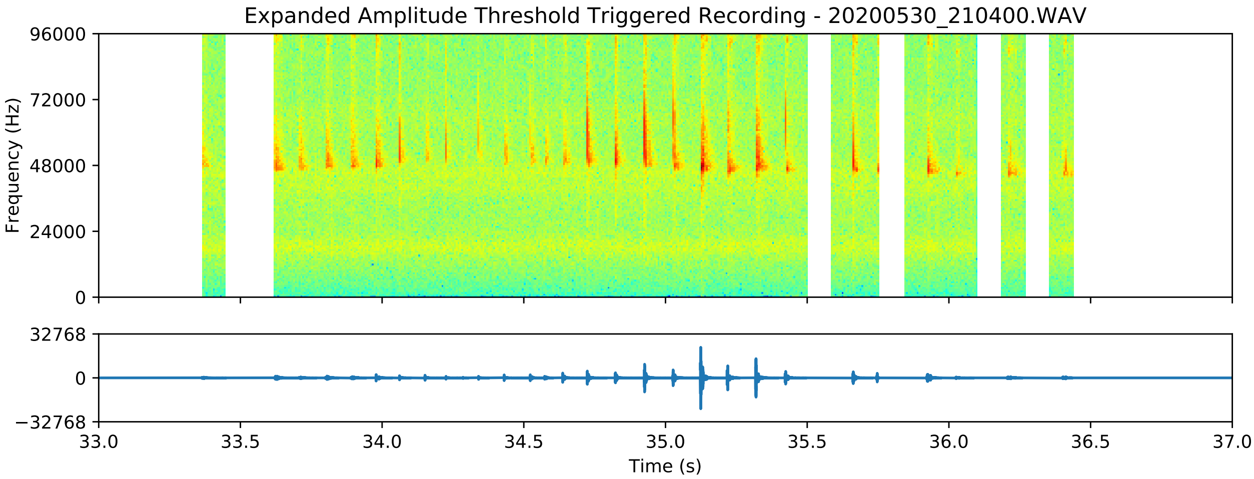

Figure 14: *Expanded* 20200530_210400.WAV *file zoomed in over the bat pass.*

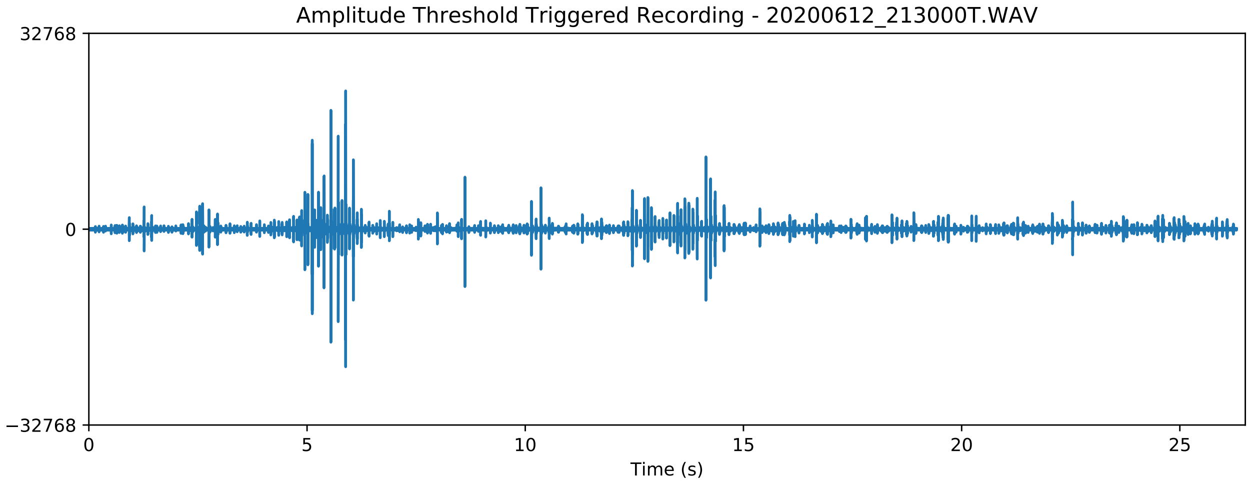

Figure 15: *Original* 20200612_213000T.WAV *file.*

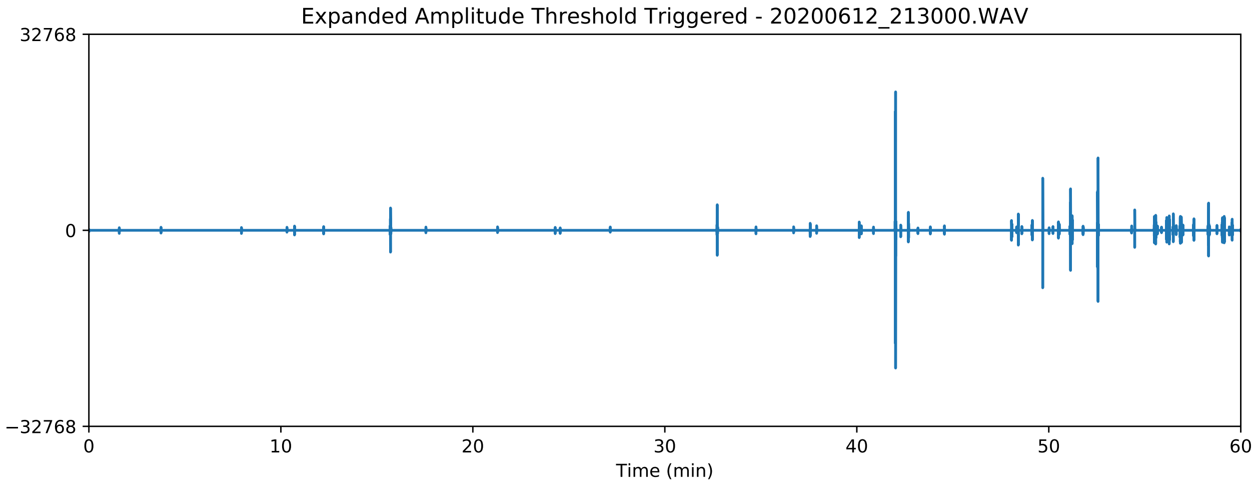

Figure 16: *Expanded* 20200612_213000.WAV *file.*

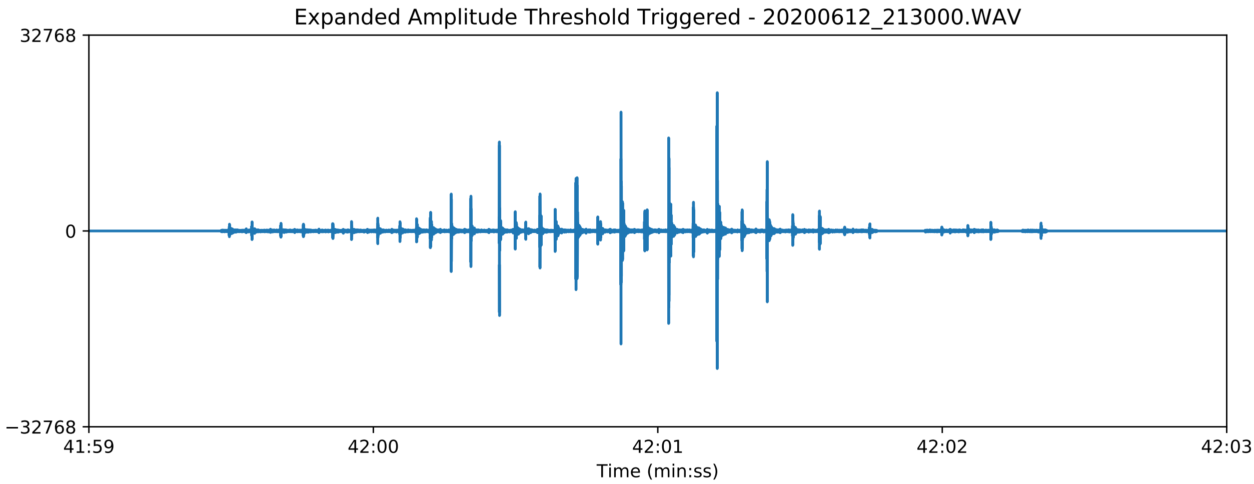

Figure 17: *Expanded* 20200612_213000.WAV *file zoomed in over the bat pass.*

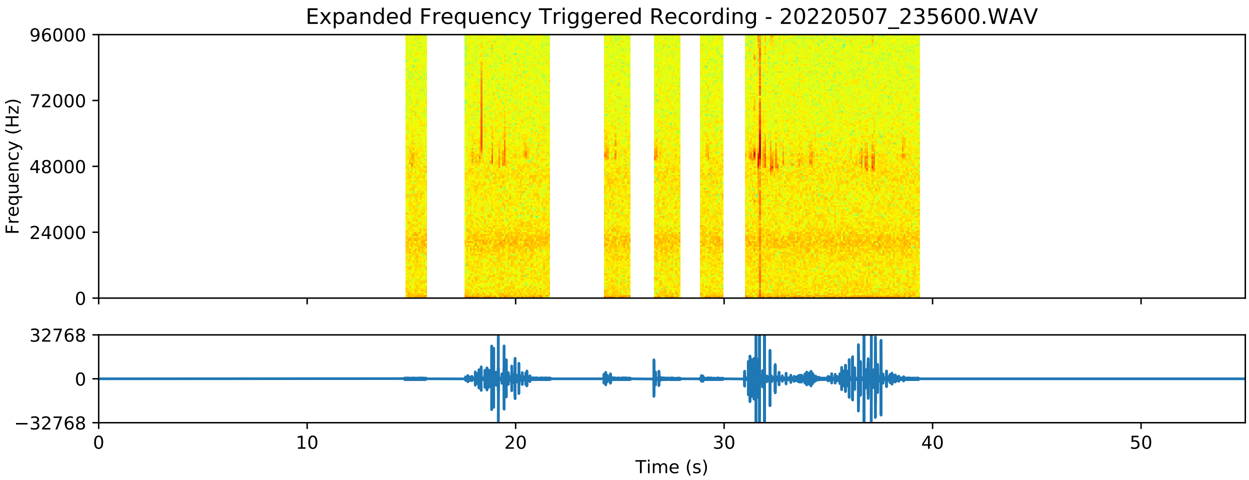

Figure 18: *Expanded* 20220507_235600T.WAV *file.*

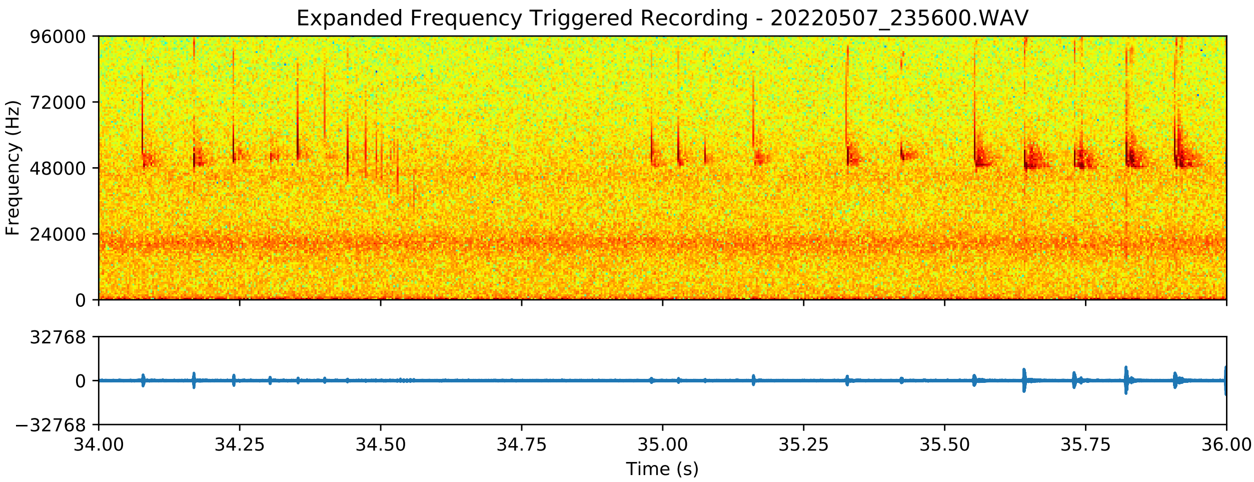

Figure 19: *Expanded* 20220507_235600T.WAV *file zoomed in over the bat feeding buzz.*

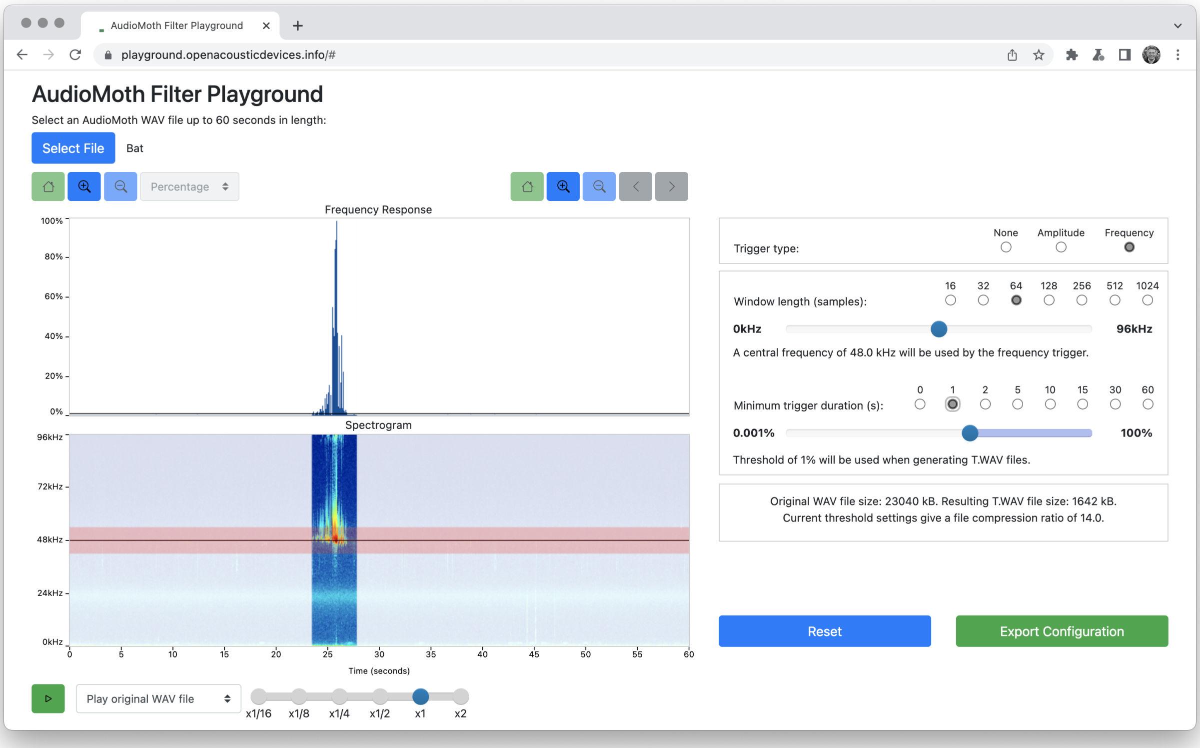

Figure 20: *AudioMoth filter playground showing an example bat recording with a frequency trigger set.*