

ANALYSIS OF ZERO-LEVEL SAMPLE PADDING OF AAC AND WMA ENCODERS

by

JEREMY YANCEY

B.S., University of Colorado Denver, 2017

A thesis submitted to the  
Faculty of the Graduate School of the  
University of Colorado in partial fulfillment  
of the requirements for the degree of  
Master of Science  
Recording Arts Program

2019

This thesis for the Master of Science degree by

Jeremy Yancey

has been approved for the

Recording Arts Program

by

Catalin Grigoras, Chair

Jeff Smith

Cole Whitecotton

Date: December 14, 2019

Yancey, Jeremy (M.S., Recording Arts Program)

Analysis of Zero-Level Sample Padding in AAC and WMA Encoders

Thesis directed by Associate Professor Catalin Grigoras

### **ABSTRACT**

During the course of the audio compression process, the codec that is used will pad the beginning of an audio file with zero-level samples. Upon playback, the zero-level samples (ZLS) are read back as absolute silence. The number of ZLS varies by which codec was used, but typically each re-compression of a file will add more ZLS to the beginning of the file. By creating multiple generations of audio files using various audio editors, this this paper hopes to shed insight of how each audio editor/codec pads files over the course of several re-compressions. The purpose of the study is to observe and note the differences in ZLS between the different codecs and audio editors across several generations of recompression and note any unique patterns across the ZLS that are added between generations within the same audio editor. In addition, the paper aims to gain a better understanding of how each program and generation affect the ZLS compared to the original audio file. With the study, we hope to use the data collected to assist in testing regarding the authenticity of an original file and use the data alongside other testing methods to determine how many times a file has been edited, recompressed, and which audio editor the edits were made in.

The form and content of this abstract are approved. I recommend its publication.

Approved: Catalin Grigoras

This is dedicated to my family and friends. Thank you, truly, for your unconditional love.

## **ACKNOWLEDGEMENTS**

I'd like to thank Jeff Smith and Catalin Grigoras for their passion for teaching. This has easily been the most fun, challenging, and rewarding two and a half years of my life. You have both continued to inspire my interest in forensics and my learning as a whole. I am thankful for the National Center for Media Forensics for the opportunities and experiences it has gifted me, as well as the people that it has introduced into my life. I am forever grateful.

# TABLE OF CONTENTS

## CHAPTER

I. INTRODUCTION.....	1
Terminology.....	1
Lossy Compression in Audio.....	2
A Brief History of the WMA Codec.....	2
A Brief History of the AAC Codec.....	2
Zero-Level Sample Padding.....	3
Purpose of the Study.....	4
II. MATERIALS AND METHODS.....	5
III. RESULTS.....	8
Adobe Audition CC 2018 AAC.....	10
Adobe Audition CC 2018 WMA.....	11
Audacity AAC.....	12
Audacity WMA.....	13
FFMPEG AAC.....	14
FFMPEG WMA.....	15
Freemake Audio AAC.....	16

Freemake Audio WMA.....	17
Garageband AAC.....	18
Apple iTunes AAC.....	19
Switch Converter AAC.....	20
Switch Converter WMA.....	21
Studio One AAC.....	22
All AAC Codecs.....	23
All WMA Codecs.....	24
IV. POST-TESTING ANALYSIS.....	25
Initial Device and Format.....	29
V. CONCLUSION.....	30
VI. FUTURE RESEARCH.....	33
REFERENCES/BIBLIOGRAPHY.....	34

## LIST OF TABLES

### TABLE

1: Terminology.....	1
2: Programs Used.....	6
3: Initial Device Data.....	8
3.1: Adobe Audition CC 2018 AAC.....	10
3.2: Adobe Audition CC 2018 WMA .....	11
3.3: Audacity AAC.....	12
3.4: Audacity WMA.....	13
3.5: FFMPEG AAC .....	14
3.6: FFMPEG WMA .....	15
3.7: Freemake Audio AAC .....	16
3.8: Freemake Audio WMA .....	17
3.9: Garageband AAC.....	18
3.10: Apple iTunes AAC .....	19
3.11: Switch Converter AAC.....	20
3.12: Switch Converter WMA .....	21
3.13: Studio One AAC.....	22



## LIST OF FIGURES

### FIGURE

3.1: Adobe Audition CC 2018 AAC – Device Averages.....	10
3.2 Adobe Audition CC 2018 WMA – Device Averages.....	11
3.3: Audacity AAC – Device Averages.....	12
3.4: Audacity WMA – Device Averages.....	13
3.5: FFMPEG AAC – Device Averages.....	14
3.6: FFMPEG WMA – Device Averages.....	15
3.7: Freemake Audio AAC – Device Averages.....	16
3.8: Freemake Audio WMA – Device Averages.....	17
3.9: Garageband AAC – Device Averages.....	18
3.10: Apple iTunes AAC – Device Averages.....	19
3.11: Switch Converter AAC – Device Averages.....	20
3.12: Switch Converter WMA – Device Averages.....	21
3.13: Studio One AAC – Device Averages.....	22
3.14: All AAC Codecs.....	23
3.15: All WMA Codecs.....	24
4.1: Table Results for Adobe Audition CC 2018.....	26

4.2: Table Results for FFMPEG AAC.....27

5.1: Decoded Text from Hex Data for File Re-compressed in Apple iTunes.....31

5.2: Spectrogram of an Original Audio File Recorded on a Tascam-DR05.....32

5.3: Spectrogram of an Audio File that has been Re-compressed Using Freemake Audio.....32

## CHAPTER I

### INTRODUCTION

#### Terminology

**Table 1: Terminology**

Term	Definition
ZLS	Zero-level Samples.
Generation	The number of times an audio files has been re-compressed.
Average	Average amount of ZLS added for that generation.
Minimum	The minimum amount of ZLS that were detected among all files.
Maximum	The maximum amount of ZLS that were detected among all files.
Mean	The average ZLS among all files.
Median	The median value of ZLS between all files.
Mode	The number of ZLS that occurred most often.
Standard Deviation	A quantity calculated to indicate the extent of deviation for the ZLS of the files as a whole.

## **Lossy Compression in Audio**

Lossy compression is a class of data encoding that partially discards data in the original content, resulting in reduced data size for storage, handling, and transmitting, at the cost of fidelity. Audio can often be compressed at 10:1 with almost imperceptible loss of quality, resulting in file sizes that are 10% of the original. The algorithms that are involved in the compression rely on psychoacoustics to reduce or to eliminate information that the algorithm deems to be redundant, taking advantage of the limitations of human hearing to create a new, smaller audio file with imperceptible change.

### **A Brief History of the WMA Codec**

The first WMA codec was created by the Windows Media team at Microsoft based on the early work of Brazilian engineer and signal processor, Henrique S. Malvar. The team at Microsoft made claims that the WMA codec could produce file sizes of half that of the widely popular MP3 codec while maintaining equivalent quality of the audio file. The claim was rejected by some. Newer versions of WMA became available which included Windows Media Audio 2 in 1999, Windows Media Audio 7 in 2000, Windows Media Audio 8 in 2001, and Windows Media Audio 9 in 2003. Microsoft first announced its plans to license WMA technology to third parties in 1999. Early versions of Windows Media Player were able to play WMA files, but backwards compatibility of the codecs was not introduced until version 9.0.

### **A Brief History of the AAC Codec**

AAC was designed to be the successor of the MP3 format, achieving better sound quality at the same bit rate. In 1972, electrical engineer Nasor Ahmed proposed the discrete cosine transform (DCT), a type of transform coding for lossy compression. This led to the development of the modified discrete cosine transform (MDCT), proposed by J. P. Princen, A. W. Johnson

and A. B. Bradley in 1987. AAC uses a purely MDCT algorithm, giving it higher compression efficiency than MP3. AAC was first introduced in 1997 and made significant improvements over the MP3 format including, higher sample rate, higher efficiency and simpler filter bank, higher coding efficiency for stationary signals, higher coding accuracy for transient signals, and much better handling of audio frequencies above 16 kHz.

### **Zero-Level Sample Padding**

Part of the coding/decoding process for lossy compression formats is to pad the newly created files with zero-level samples (ZLS). When read back, these ZLS are interpreted as absolute silence in the file. Depending on which codec and file specifications were used, there is a variable amount of ZLS that are added. Generally, each time a file is compressed, more ZLS are added to the file, resulting in a file that is longer than the original, with a greater period of silence at the beginning of the file.

A number of causes have been noted as to why this occurs when a file is compressed<sup>1</sup>. Digital audio files are processed in blocks which are processed based on a number of audio samples. The algorithm of the codec cannot start until a signal buffer of samples has been filled. This can be required along the length of the transmission chain, leading to the first cause for the need of ZLS in the audio file.

The second cause for ZLS is the use of frequency-domain processing. All signals have to go through a filter-bank. Any encoder/decoder analysis and synthesis filter-bank system leads to a signal delay of samples.

The third cause of ZLS is the need for an amount of time for a look-ahead time in the signal. Some encoders require this for the algorithm to decide and implement strategies that are

needed for the internal processing of the data. The actual processing of a block of audio data takes time. Real-time hardware implementations mostly choose to add another delay of samples to give the algorithm more time to make decisions.

### **Purpose of the Study**

The purpose of the study is to observe and note the differences in ZLS between the different codecs and audio editors across several generations of recompression. The paper hopes to shed insight to see if there are any unique patterns across the ZLS that are added between generations within the same audio editor and gain a better understanding of how each program and generation affect the ZLS compared to the original audio file. With the study, we hope to use the data collected to assist in testing regarding the authenticity of an original file and use the data alongside other testing methods to determine how many times a file has been edited, recompressed, and which audio editor the edits were made in.

## CHAPTER II

### MATERIALS AND METHODS

The procedure of the study was to take various audio recordings, record 30 second audio files, recompress the audio files using various audio editors, and then analyze the amount of ZLS that were added to the beginning of the file. The recordings that were used for analysis originated from the following devices:

Zoom H-1n

Tascam DR-05

Tascam DR-22

Olympus Ws-852

Olympus Ws-802

iPhone Xs Voice Memos

These devices were chosen as they represented audio recorders that were easily accessible to the general public. The devices each had different saved file types, with the exception of the Tascam Dr-05 and Tascam-Dr22 which both saved as mono .WAV files. The Zoom H-1n saved a stereo .MP3 file, the Olympus Ws-852 saved a stereo .WAV file, The Olympus Ws-802 saved a mono .WMA file, and the iPhone voice memo saved a mono .AAC file. This was done to see if there were any differences between in the amount of ZLS added based on the originating file format.

Each device recorded twelve (12) files. The files were recorded in 3 different ambient environments:

(4) Loud music

(4) Moderate noise

(4) Low level ambient noise

Once all files were recorded and gathered, the next step in the procedure was to recompress the files. The files were each brought into the audio program under examination and compressed with the default settings for both (if applicable) WMA and AAC formats. The newly created file was then brought back into the program and the steps were repeated. This process was continued until a 4<sup>th</sup> new generation of the original audio file was created.

Once the new files had been created, they were analyzed by a MATLAB script that detected and reported back the number of zero-level samples in each channel before there was any change in signal. All audio files were recorded at 16bit/128kbps with the exception of Apple Voice Memos on the iPhone Xs which was recorded at 64kbps.

The following programs were used to create the files for analysis:

**Table 2: Programs Used**

Program	Program Version	AAC Codec Tested	WMA Codec Tested
Adobe Audition CC 2018	11.1.1.3	Yes	Yes
Audacity	2.2.2	Yes	Yes
Ffmpeg	4.2.1	Yes	Yes
Freemake Audio	1.1.8.19	Yes	Yes



Garageband	10.0.3	Yes	N/A
iTunes	12.10.2.3	Yes	N/A
Studio One	4.5.4.54067	Yes	N/A
Switch Converter	7.33	Yes	Yes

The only settings that were modified upon exporting and compressing the file was the bitrate. The bitrate settings were changed to 128kbps to be consistent with the original settings of the audio recording devices, minus the iPhone Voice Memos.

## CHAPTER III

### RESULTS

The following section describes the results from the data collection. For each software that was tested, there is a table that includes, the average, minimum, maximum, median, mode, and standard deviation of zero-level samples for all devices tested within that software, for each new generation created. Additionally, there is a graph that shows the average amount of zero-level samples across the generations for each device that was tested within the program. The following table displays the devices used, along with the original file format, minimum zero-level samples of the original files, maximum, average, and standard deviation.

**Table 3: Initial Device Data**

Device	Format	Minimum	Maximum	Average	Standard Deviation
Zoom H-1n	.WAV	138	8700	2916	2814
Tascam DR-05	.WAV	131	12446	5620	4651
Tascam DR-22	.WAV	0	0	0	0
Olympus Ws-852	.MP3	348	32957	10216	12411

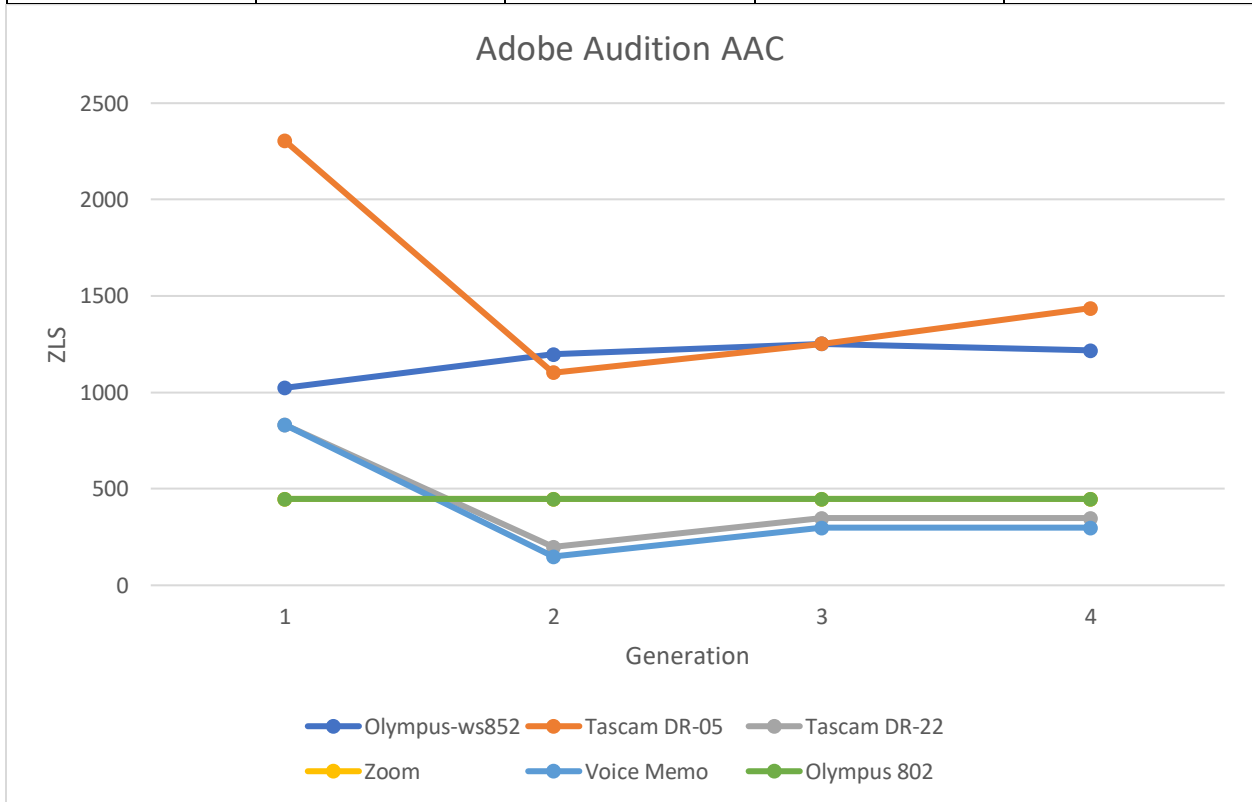
**Table 3 Continued**

Device	Format	Minimum	Maximum	Average	Standard Deviation
Olympus Ws-802	.WMA	0	0	0	0
iPhone Xs Voice Memos	.AAC	1984	1984	1984	0

**Adobe Audition CC 2018 AAC**

**Table 3.1: Adobe Audition CC 2018 AAC**

Generation	I	II	III	IV
Average	981	591	674	699
Minimum	0	0	0	0
Maximum	1024	3008	2048	2942
Median	832	448	448	448
Mode	832	448	448	448
Standard Deviation	2328	559	503	643

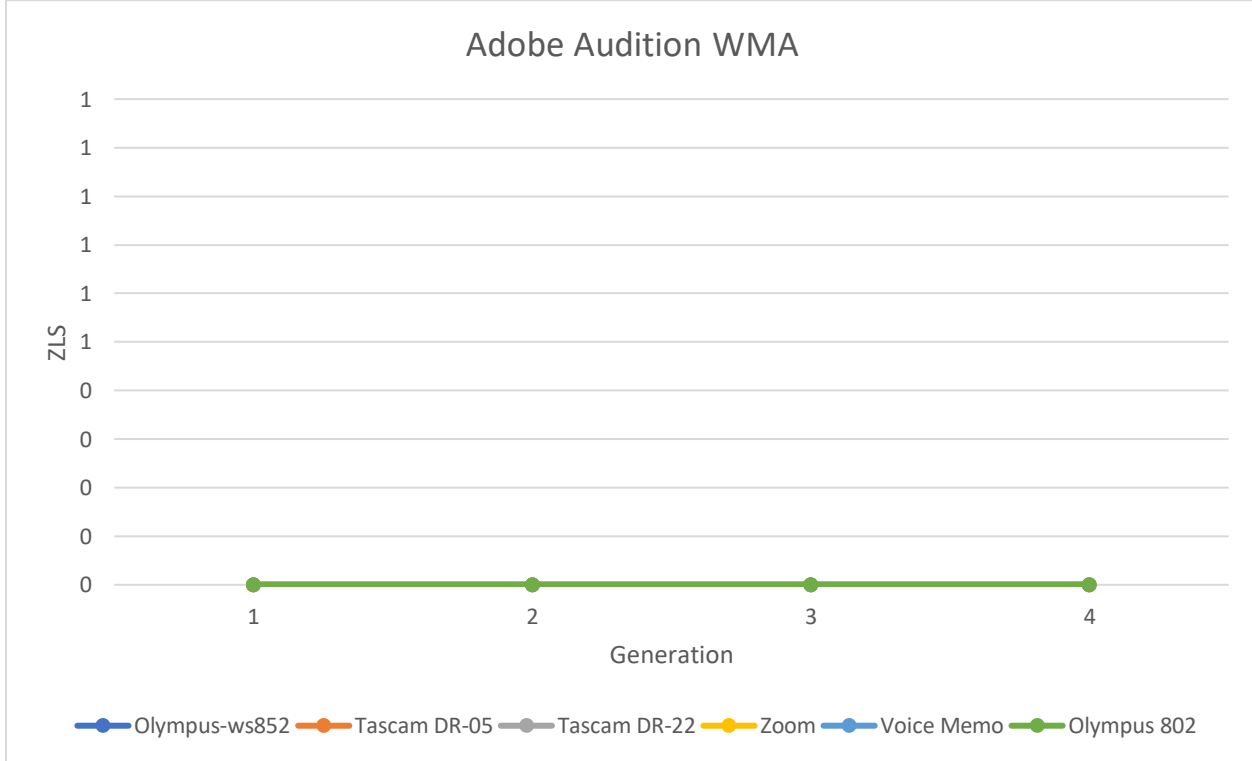


**Figure 3.1: Adobe Audition CC 2018 AAC – Device Averages**

**Adobe Audition CC 2018 WMA**

**Table 3.2: Adobe Audition CC 2018 WMA**

Generation	I	II	III	IV
Average	0	0	0	0
Minimum	0	0	0	0
Maximum	0	0	0	0
Mean	0	0	0	0
Median	0	0	0	0
Mode	0	0	0	0
Standard Deviation	0	0	0	0

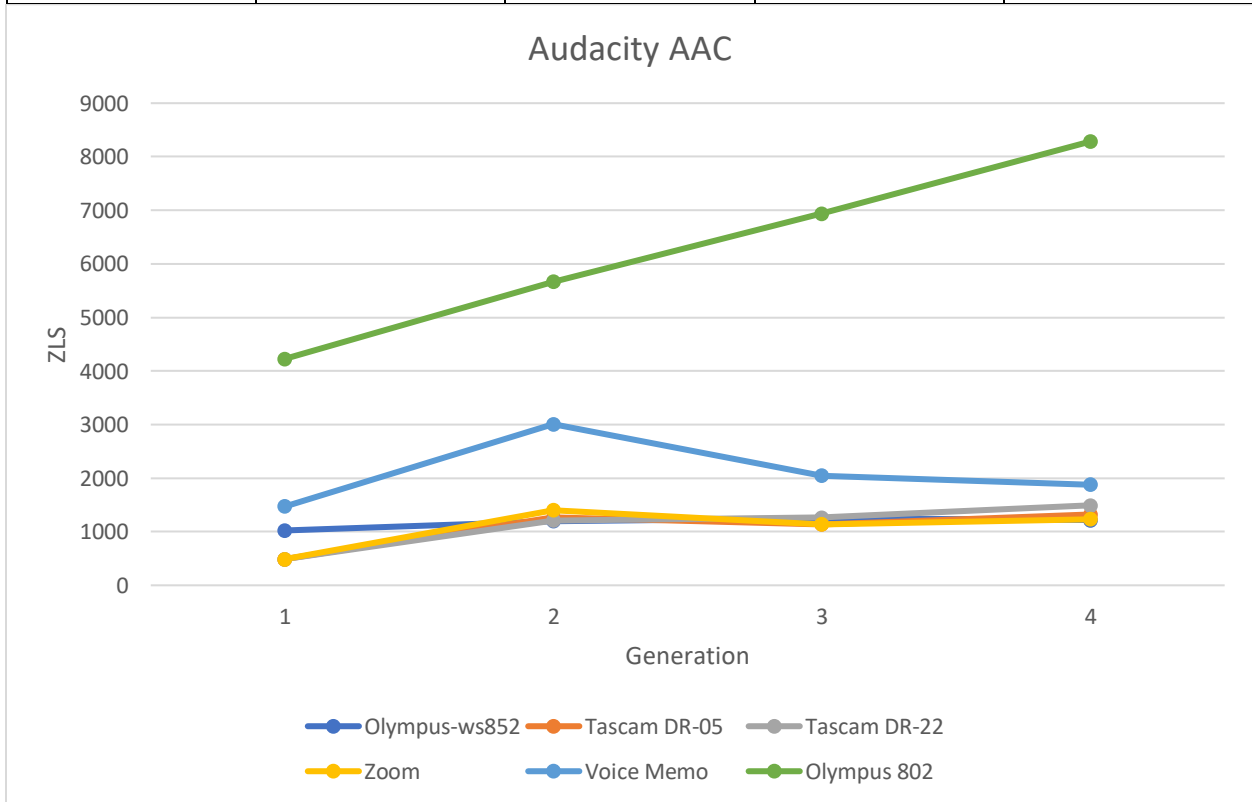


**Figure 3.2 Adobe Audition CC 2018 WMA – Device Averages**

## Audacity AAC

**Table 3.3: Audacity AAC**

Generation	I	II	III	IV
Average	1366	2292	2297	2572
Minimum	0	1024	10241	1024
Maximum	5312	7104	7168	10304
Median	1366	1408	1024	1508
Mode	0	1024	1024	1024
Standard Deviation	1513	1766	2267	2769

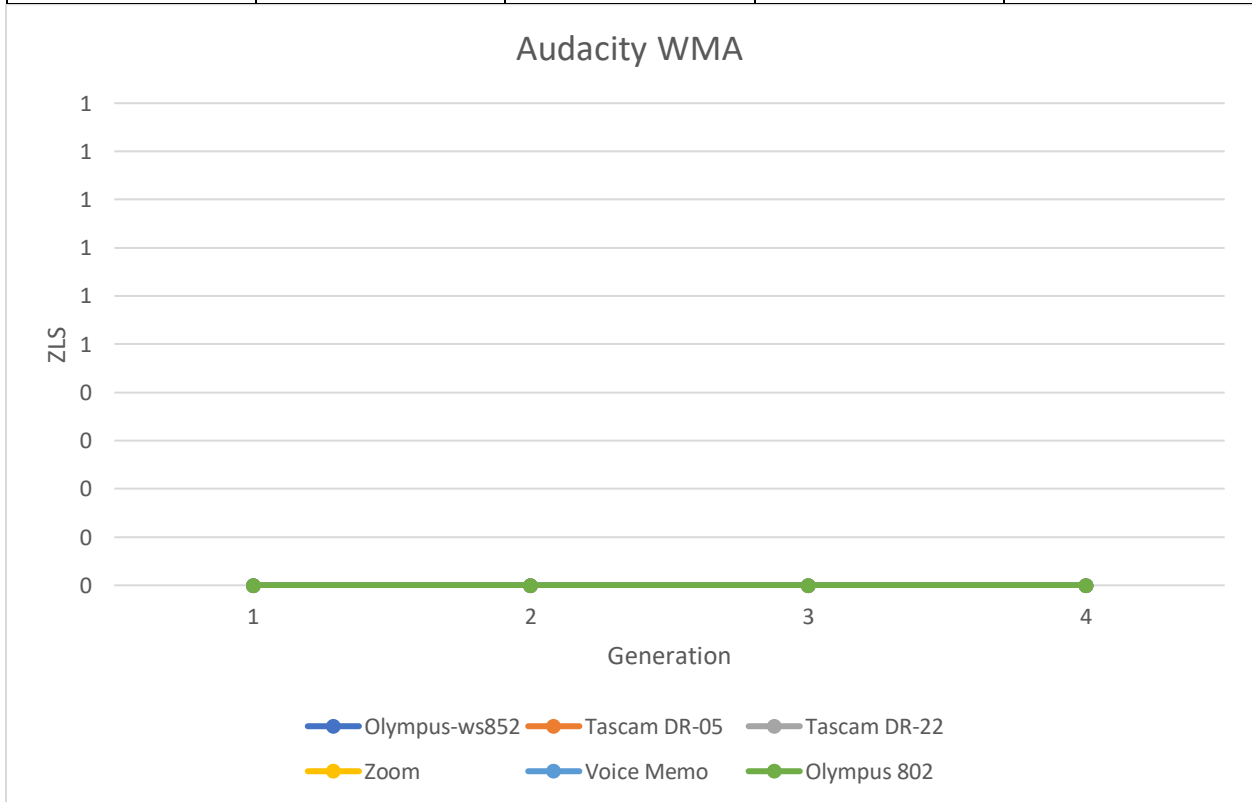


**Figure 3.3: Audacity AAC – Device Averages**

## Audacity WMA

**Table 3.4: Audacity WMA**

Generation	I	II	III	IV
Average	0	0	0	0
Minimum	0	0	0	0
Maximum	0	0	0	0
Median	0	0	0	0
Mode	0	0	0	0
Standard Deviation	0	0	0	0

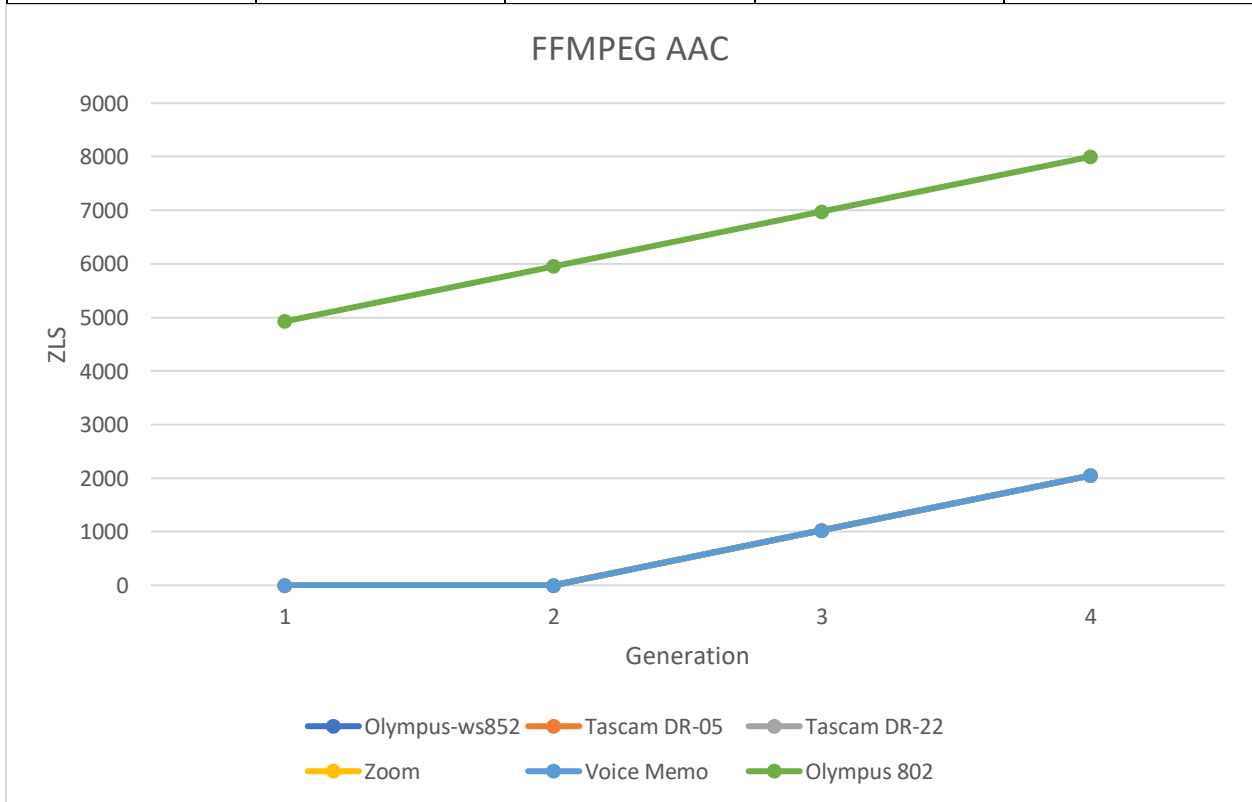


**Figure 3.4: Audacity WMA – Device Averages**

## FFMPEG AAC

**Table 3.5: FFMPEG AAC**

Generation	I	II	III	IV
Average	821	1029	2053	3077
Minimum	0	0	1024	2048
Maximum	4928	5952	6976	8000
Median	0	0	1024	2048
Mode	0	0	1024	2048
Standard Deviation	1923	2204	2204	2204



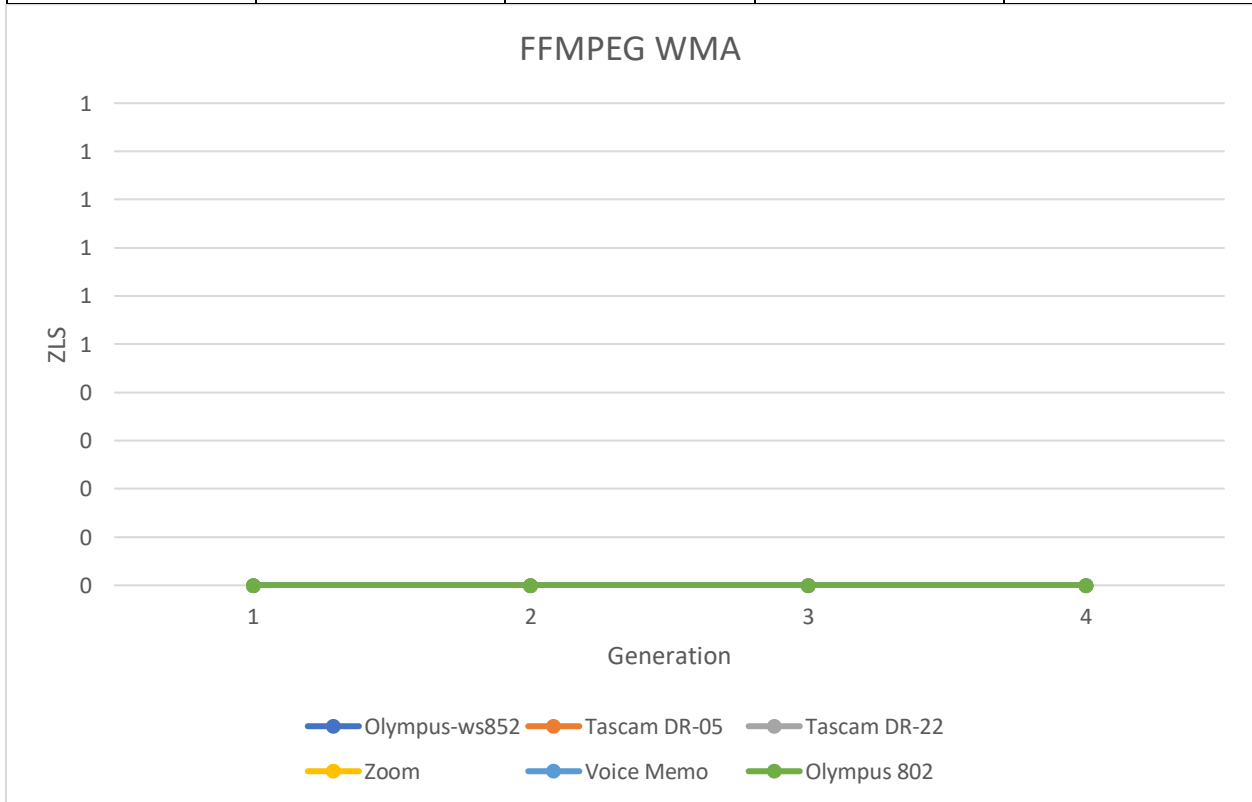
**Figure 3.5: FFMPEG AAC – Device Averages**



## FFMPEG WMA

**Table 3.6: FFMPEG WMA**

Generation	I	II	III	IV
Average	0	0	0	0
Minimum	0	0	0	0
Maximum	0	0	0	0
Median	0	0	0	0
Mode	0	0	0	0
Standard Deviation	0	0	0	

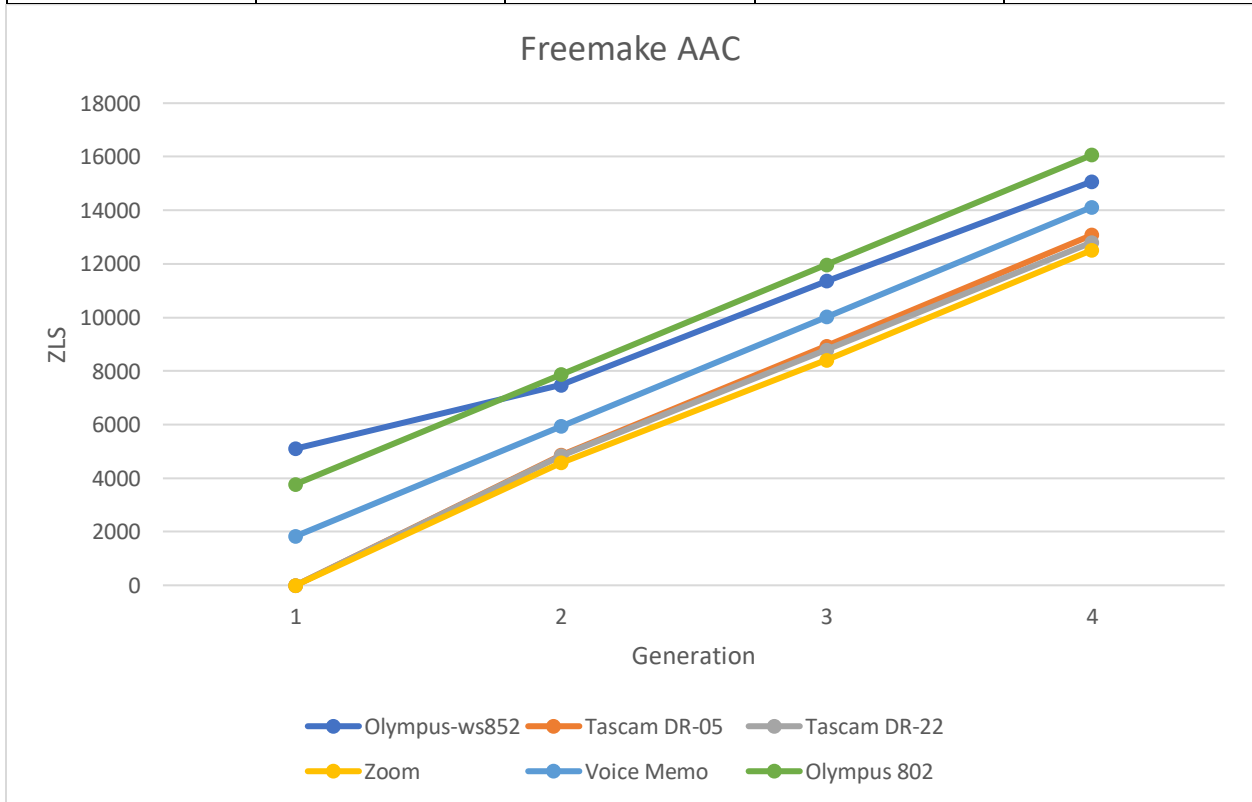


**Figure 3.6: FFMPEG WMA – Device Averages**

## Freemake Audio AAC

**Table 3.7: Freemake Audio AAC**

Generation	I	II	III	IV
Average	1786	5925	9916	13943
Minimum	0	4094	7166	11198
Maximum	18047	20095	22399	23423
Median	0	4991	9024	13188
Mode	0	4926	9022	16064
Standard Deviation	2922	2395	2251	1908

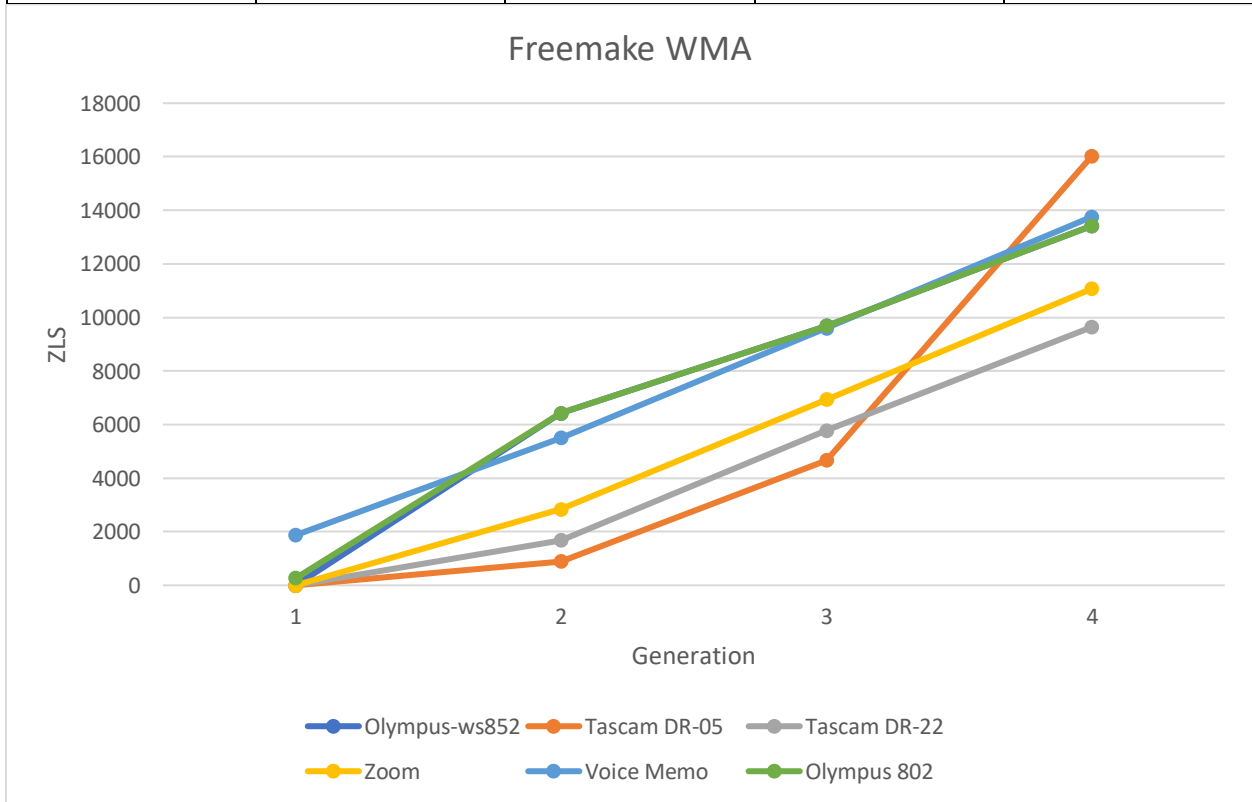


**Figure 3.7: Freemake Audio AAC – Device Averages**

## Freemake Audio WMA

**Table 3.8: Freemake Audio WMA**

Generation	I	II	III	IV
Average	357	3959	7723	12891
Minimum	0	0	4096	1892
Maximum	2395	18909	21467	81912
Median	0	3808	7904	12032
Mode	0	0	4096	12032
Standard Deviation	560	3814	3472	10667

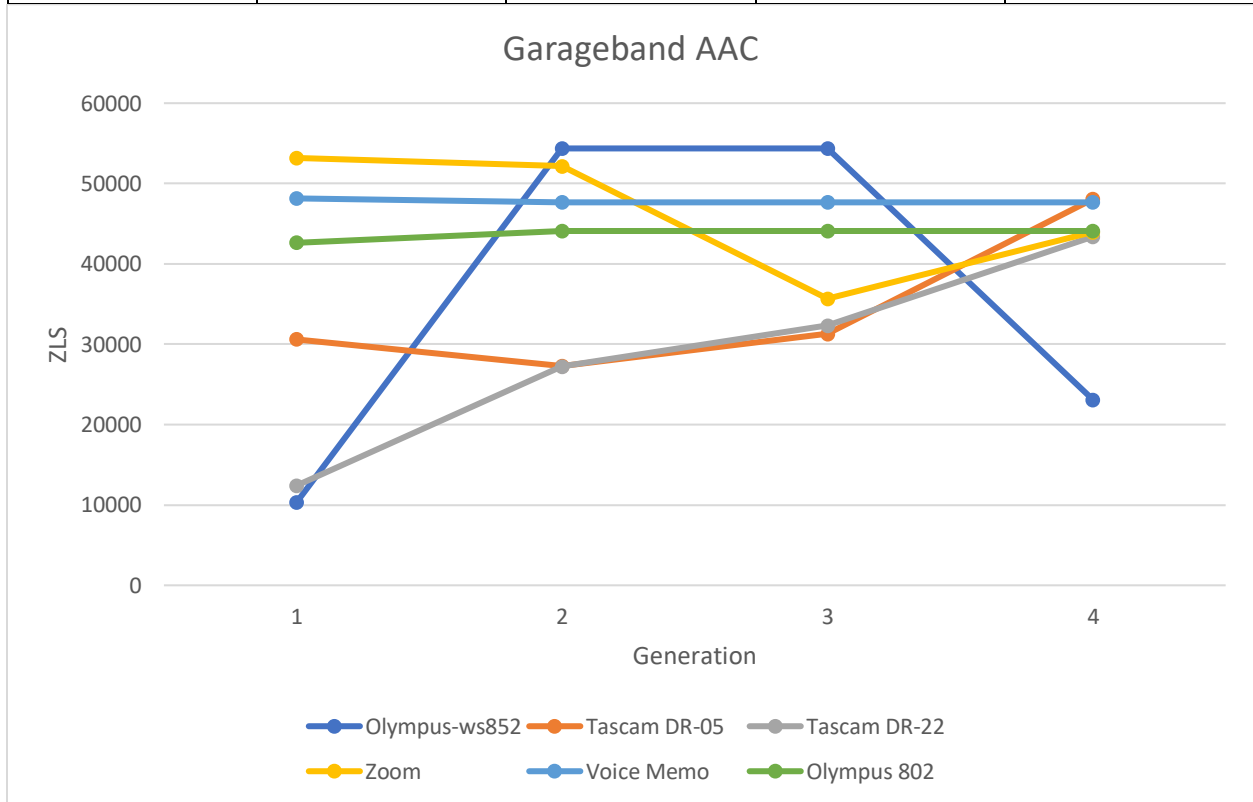


**Figure 3.8: Freemake Audio WMA – Device Averages**

## Garageband AAC

**Table 3.9: Garageband AAC**

Generation	I	II	III	IV
Average	32888	42133	40901	41713
Minimum	9151	21567	28415	12671
Maximum	67071	65215	64191	65471
Median	39615	50623	40635	44031
Mode	58183	52159	34175	33535
Standard Deviation	17641	17839	11207	13851

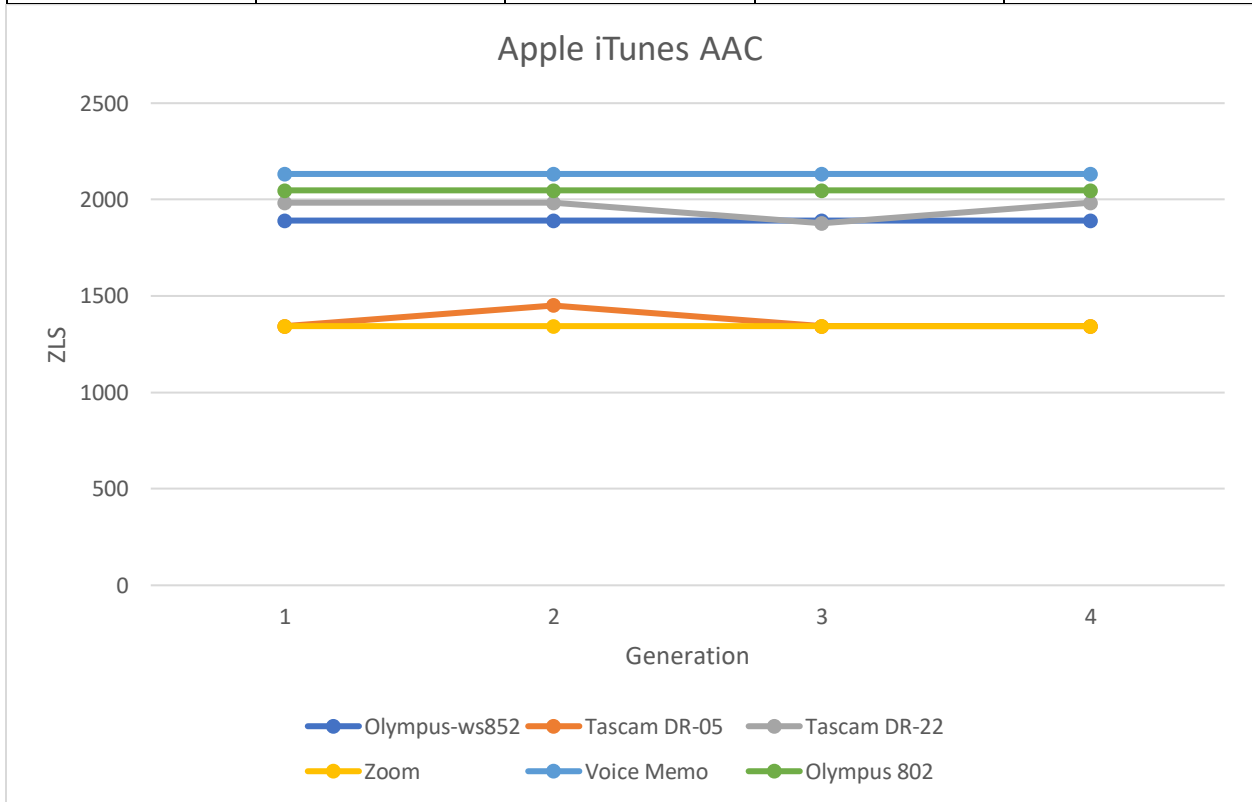


**Figure 3.9: Garageband AAC – Device Averages**

## Apple iTunes AAC

**Table 3.10: Apple iTunes AAC**

Generation	I	II	III	IV
Average	1791	1809	1773	1791
Minimum	1024	1024	1024	1024
Maximum	2559	2559	2559	2559
Median	1984	1984	1984	1984
Mode	1984	1984	1984	1984
Standard Deviation	435	428	446	435

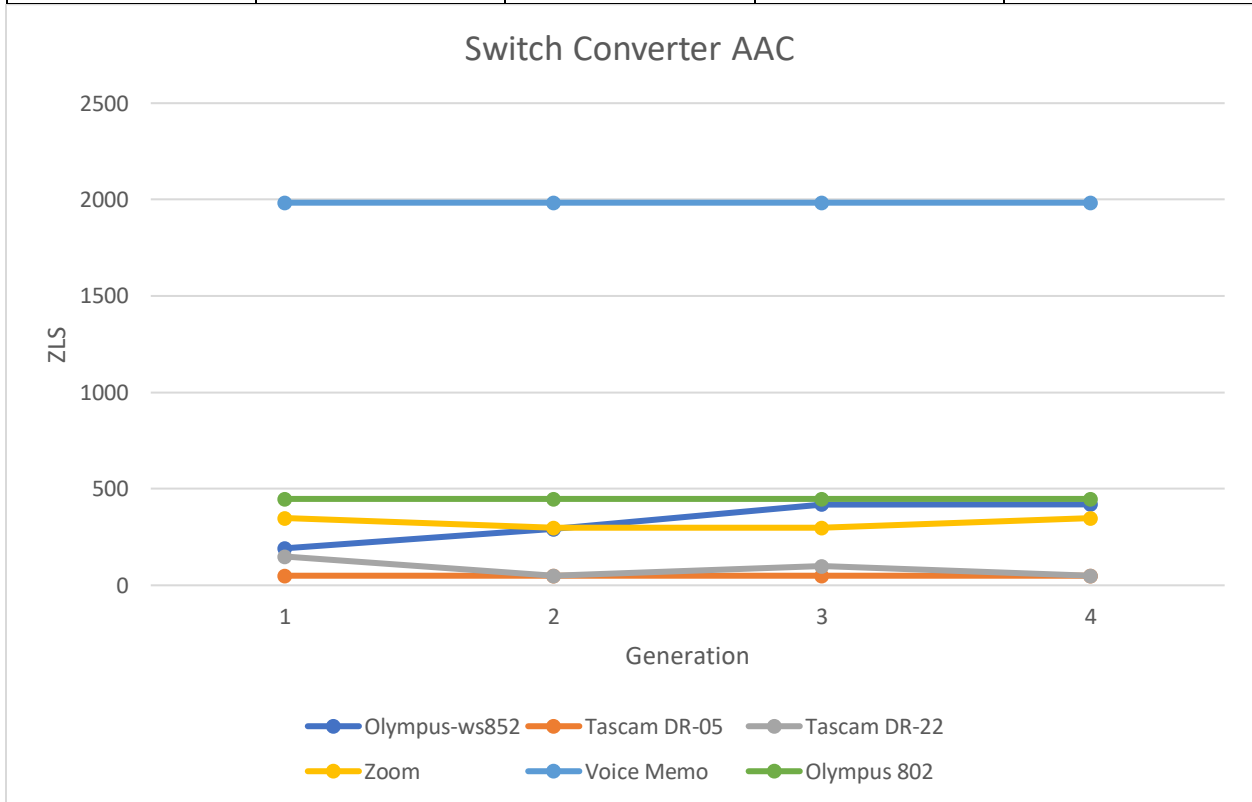


**Figure 3.10: Apple iTunes AAC – Device Averages**

## Switch Converter AAC

**Table 3.11: Switch Converter AAC**

Generation	I	II	III	IV
Average	529	520	550	550
Minimum	0	0	0	0
Maximum	1984	1984	1984	1984
Median	448	448	448	448
Mode	0	0	0	0
Standard Deviation	490	484	565	565

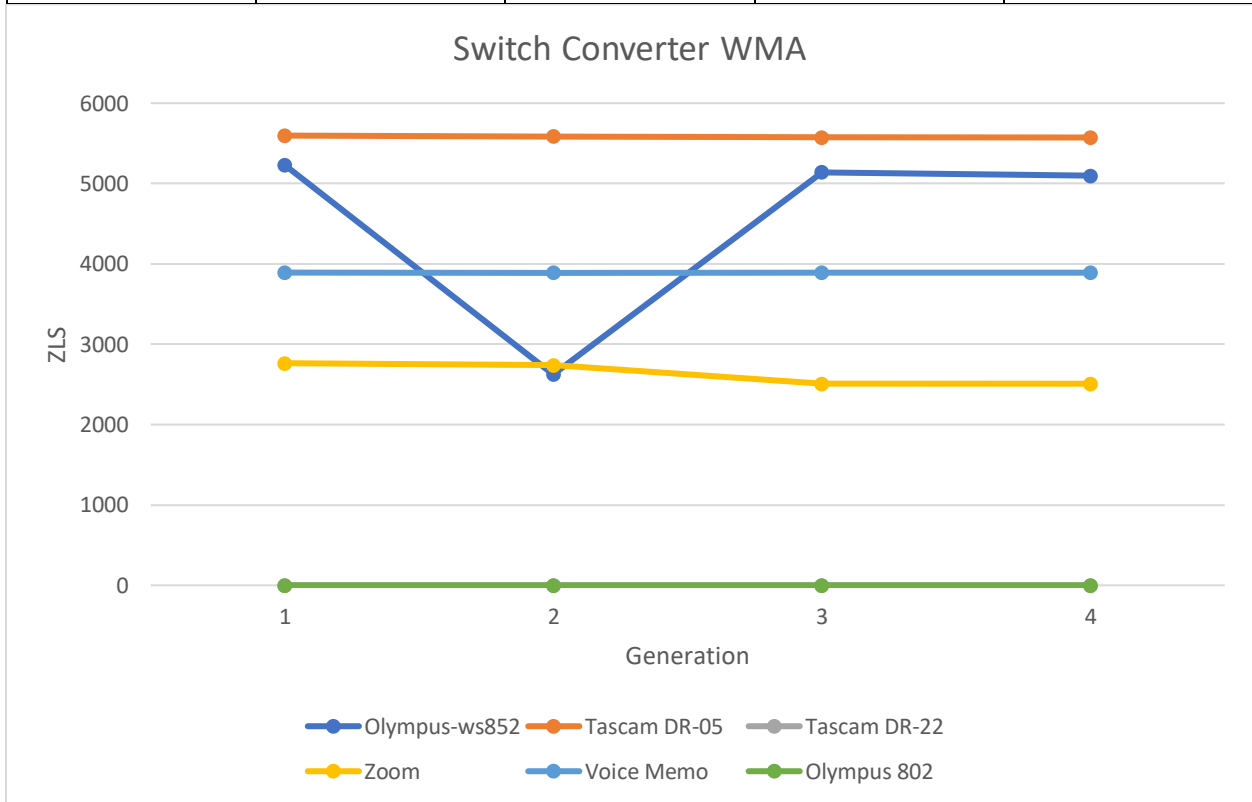


**Figure 3.11: Switch Converter AAC – Device Averages**

## Switch Converter WMA

**Table 3.12: Switch Converter WMA**

Generation	I	II	III	IV
Average	2913	2473	2853	2845
Minimum	0	0	0	0
Maximum	12032	12446	12944	12900
Median	456	127	384	362
Mode	0	0	0	0
Standard Deviation	4019	3317	3974	3968

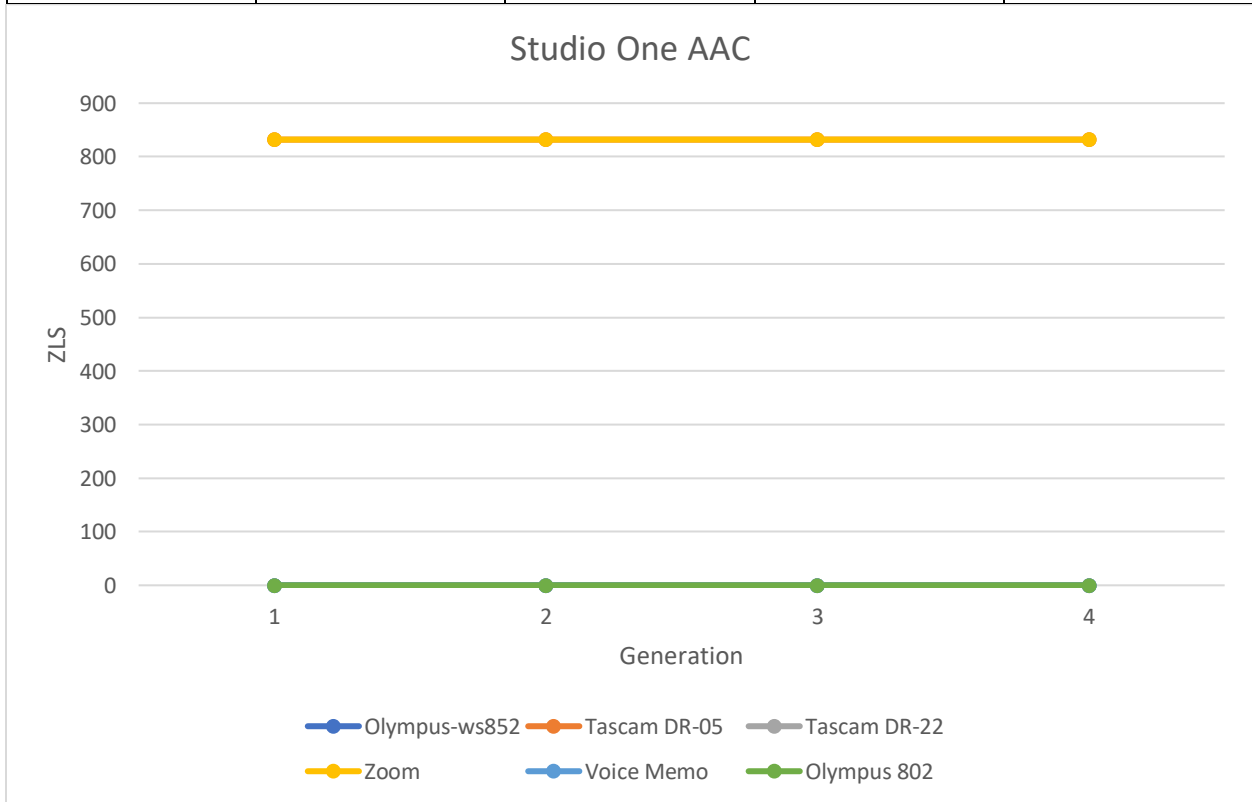


**Figure 3.12: Switch Converter WMA – Device Averages**

## Studio One AAC

**Table 3.13: Studio One AAC**

Generation	I	II	III	IV
Average	277	277	277	277
Minimum	0	0	0	0
Maximum	832	832	832	832
Median	0	0	0	0
Mode	0	0	0	0
Standard Deviation	403	392	392	392

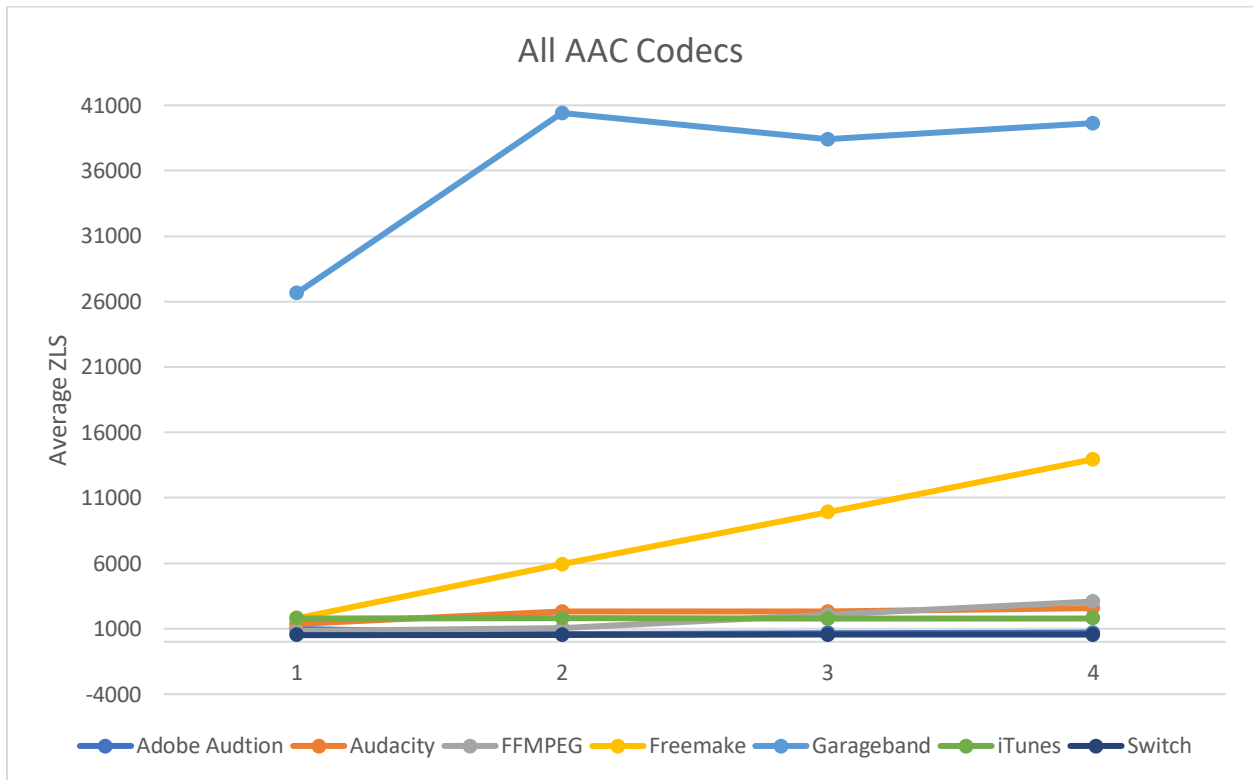


**Figure 3.13: Studio One AAC – Device Averages**



## All AAC Codecs

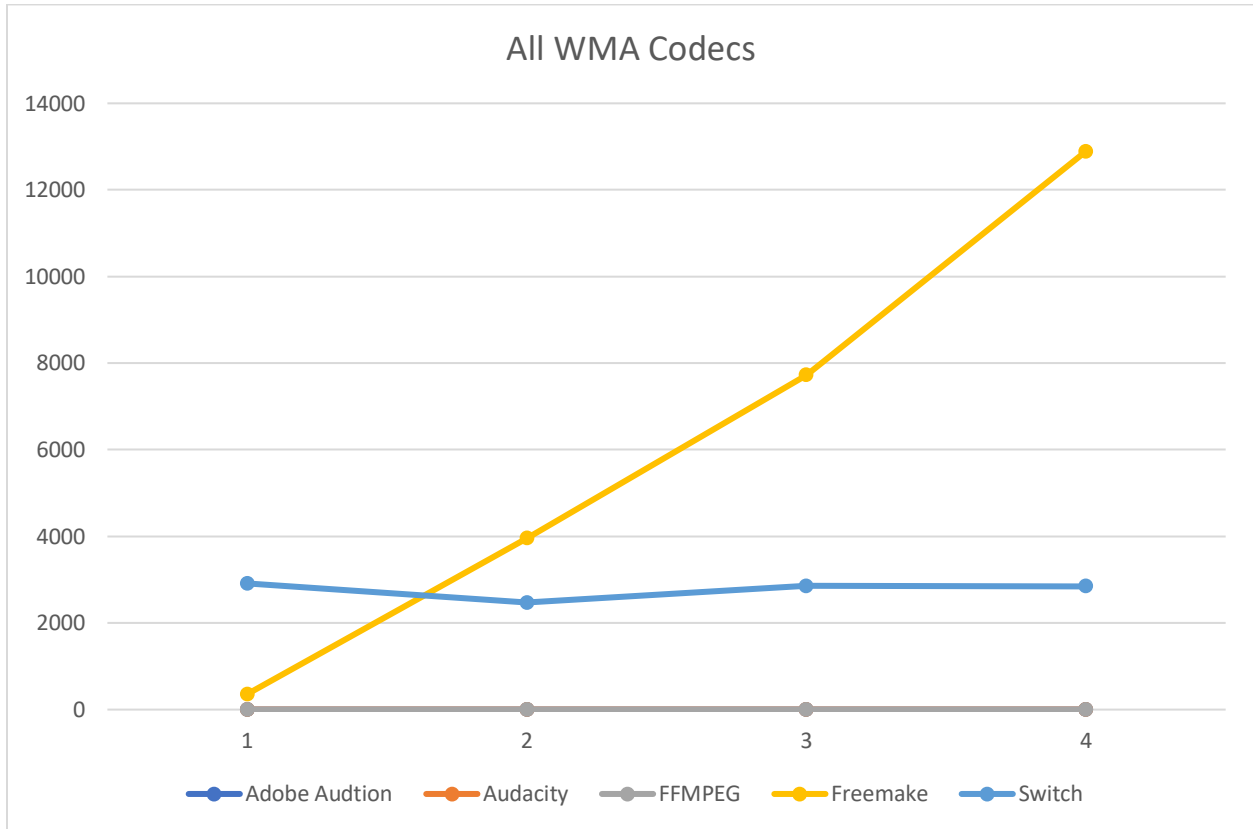
This section compiles all the data from all the audio editing programs where the .AAC codec was used.



**Figure 3.14: All AAC Codecs**

## All WMA Codecs

This section compiles all the data from all the audio editing programs where the .WMA codec was used.



**Figure 3.15: All WMA Codecs**

## CHAPTER IV

### POST-TESTING ANALYSIS

After analyzing the number of zero-level samples for the different audio devices, audio editors, and audio generations, there are patterns that are noted for several of the different programs. There were not always expected linear growth in the zero-level samples when examining the averages across device and program audio generations as a whole, but when looking at an individual audio file and its generations in a specific program, there are several repeated sequence numbers. In some programs, this made analyzing the sequences show more of a pattern rather than when only examining the average across generations for all of the audio files.

An example of this would be the files that were re-compressed in Adobe Audition CC 2018. When examining the averages from the line graph, there does not appear to be a pattern in the zero-level samples that are added. When looking at the table that the graph is pulling data from, with the exception of one device (Tascam DR-05), every file had either 0, 448, or 832 zero-level samples added.

Tascam DR-22			
Gen	Loud Music 1	Loud Music 2	Loud Music 3
2	832	832	832
3	448	0	0
4	448	0	448
5	448	0	448
Zoom			
Gen	Loud Music 1	Loud Music 2	Loud Music 3
2	448	448	448
3	448	448	448
4	448	448	448
5	448	448	448
Voice Memo			
Gen	Loud Music 1	Loud Music 2	Loud Music 3
2	832	832	832
3	448	0	0
4	448	0	448
5	448	0	448
Olympus 802			
	Loud Music 1	Loud Music 2	Loud Music 3
2	448	448	448
3	448	448	448
4	448	448	448
5	448	448	448

**Figure 4.1: Table Results for Adobe Audition CC 2018**

When examining the .AAC files made in FFMPEG, with the exception of the Olympus Ws-802, each device had no new zero-level samples added in the first two generations. In the last two generations, there were exactly 1,024 and 2,048 zero-level in total. The Olympus Ws-802

added 1,024 zero-level samples each generation, but had a 4,928 zero-level samples in each file of the first re-compression, rather than zero like the other devices.

	Zoom		
Gen	LM1	LM2	LM3
2	0	0	0
3	0	0	0
4	1024	1024	1024
5	2048	2048	2048

	Voice Memo		
Gen	LM1	LM2	LM3
2	0	0	0
3	0	0	0
4	1024	1024	1024
5	2048	2048	2048

	Olympus 802		
	LM1	LM2	LM3
2	4928	4928	4928
3	5952	5952	5952
4	6976	6976	6976
5	8000	8000	8000

**Figure 4.2: Table Results for FFMPEG AAC**

There were multiple programs that did not add any zeros with each subsequent generation of re-compression. The programs are as follows:

- Freemake Audio (AAC)
- Freemake Audio (WMA)
- FFMPEG (WMA)

Programs did not add any zero level samples during WMA encoding:

- Adobe Audition CC 2018 (WMA)
- Audacity (WMA)
- FFMPEG (WMA)

An explanation for both Audacity and FFMPEG both not adding any zero-level samples is that they use the same encoding. Encoding as a .WMA file is not default with Audacity and an extension of FFMPEG must be installed as an add-on to the program before encoding as a .WMA file is possible.

Programs where there was little change throughout generations:

- Apple iTunes (AAC)
- Switch Converter (AAC)
- Switch Converter (WMA)

Of the 13 tests conducted, only 2 had no discernable patterns in either the tables or averages across all devices and generations. The programs where there were no discernable patterns were as follows:

- Audacity (AAC)

- Garageband (AAC)

### **Initial Device and Format**

It could be expected that the program being used and the codec would have the most impact on the number of zero-level samples that are added to the different generation of audio files. While this seemed to be the case for many of the different programs, the different devices seemed to behave differently from each other in some programs depending on what the original audio file format was. In some programs, the Olympus Ws-802 which generates .WMA file had significantly more zero-level samples added in Audacity (AAC) and FFmpeg (AAC). When testing the Olympus Ws-802 in Switch Converter (WMA), none of the files had any zero-level samples added.

When looking at the averages for Garageband, there are no devices that follow a pattern close to any other device. Some devices increase throughout the generations, while other increase and decrease without any pattern.

## CHAPTER V

### CONCLUSION

The results of previous studies<sup>2</sup> done on other audio codecs have shown that there is not always a linear growth in the number of zero-level samples that are added to the generations of an audio file after re-compression. The results of this study are similar. Some programs behaved as expected and added zero-level samples to each generation, some programs added an initial number during the first generation of re-compression and then kept relatively the same amount in following generations, and others did not introduce any zero-level samples at all. Because of this, the analysis of zero-level samples on its own could not be used to determine the authenticity of an audio file or even the generation.

There were patterns or numbers that were seen throughout testing in certain programs that were of note. If there was not always a linear growth in the number of zero-level samples detected, there were times where the same number of zero-level samples that were added appeared.

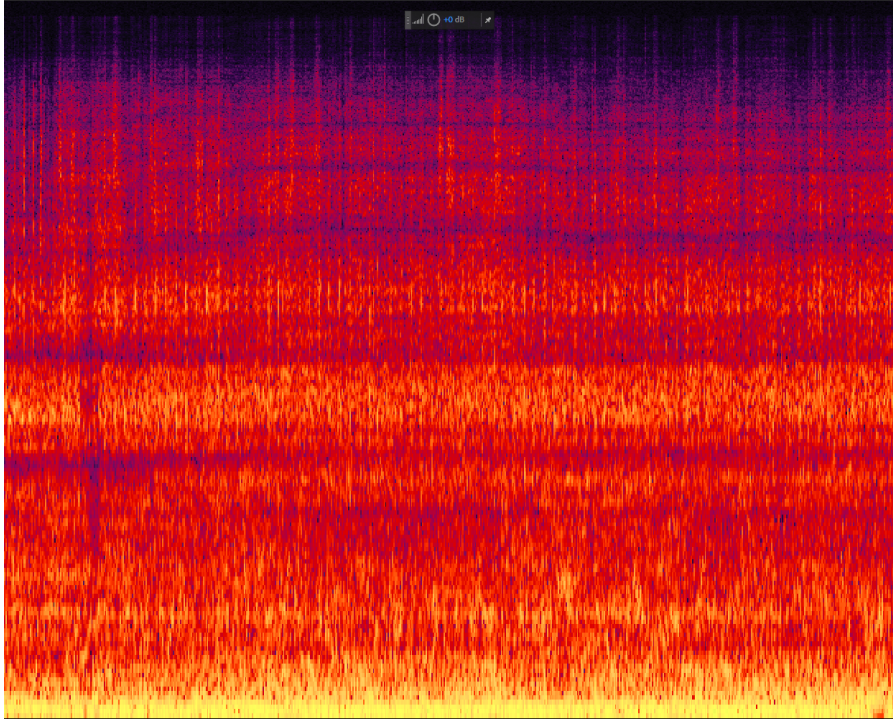
An analysis of the zero-level samples could be used in addition to other means of authenticity testing to assist in verifying the results. The testing could be used in conjunction with tests like an analysis of a file's metadata. In figure 6, there is decoded text from the hex data of a file that has been compressed in Apple iTunes. By testing the zero-level samples of the audio file and with the information from the hex code analysis, the zero-level samples can serve as second confirmation that the file is not authentic.



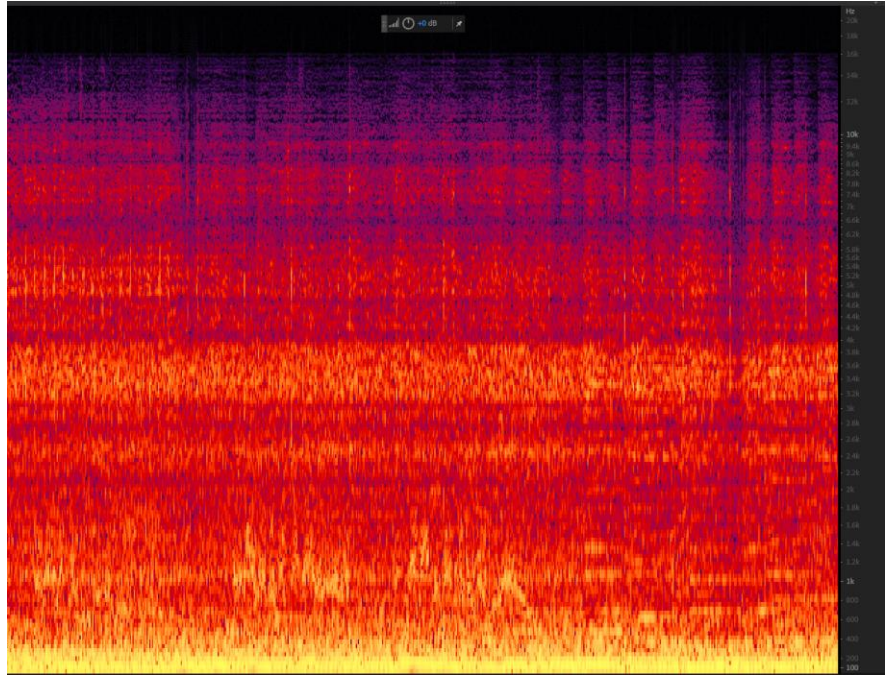
Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
000017C0	30	00	05	EC	AE	00	06	1C	D2	00	06	4C	C3	00	06	7C	0..i@...Ò...LÃ..
000017D0	65	00	06	AC	61	00	06	DC	34	00	07	0C	09	00	07	3B	e...-a...Û4.....;
000017E0	FF	00	07	6B	F2	00	07	9B	C3	00	07	CB	EF	00	07	FB	ÿ..kò...»Ã...Ëi...û
000017F0	99	00	08	2B	65	00	08	5B	46	00	08	8B	36	00	08	BB	™...+e...[F...<6...»
00001800	1E	00	08	EA	F4	00	09	1B	29	00	09	4A	ED	00	09	7A	...êô...))..Ji...z
00001810	AD	00	09	AA	92	00	09	DA	76	00	0A	0A	88	00	0A	3A	...²'...Ûv...^...:
00001820	3A	00	0A	6A	2D	00	0A	99	FA	00	0A	C9	DE	00	0A	F9	:...j-...™ú...ÉP...ù
00001830	C8	00	0B	29	BF	00	0B	59	8D	00	0B	89	6F	00	0B	B9	È...)¿...Y...%o...²
00001840	9B	00	0B	E9	32	00	0C	19	16	00	0C	48	FB	00	00	00	>...é2.....Hû...
00001850	FA	75	64	74	61	00	00	00	F2	6D	65	74	61	00	00	00	úudta...òmeta...
00001860	00	00	00	00	22	68	64	6C	72	00	00	00	00	00	00	00	...."hdr.....
00001870	00	6D	64	69	72	61	70	70	6C	00	00	00	00	00	00	00	.mdirappl.....
00001880	00	00	00	00	00	00	C4	69	6C	73	74	00	00	00	BC	2D	.....Äilst...¼-
00001890	2D	2D	2D	00	00	00	1C	6D	65	61	6E	00	00	00	00	63	---...mean....c
000018A0	6F	6D	2E	61	70	70	6C	65	2E	69	54	75	6E	65	73	00	om. <a href="#">apple</a> .iTunes.
000018B0	00	00	14	6E	61	6D	65	00	00	00	00	69	54	75	6E	53	...name....iTunS
000018C0	4D	50	42	00	00	00	84	64	61	74	61	00	00	00	01	00	MPB....„data.....
000018D0	00	00	00	20	30	30	30	30	30	30	30	30	20	30	30	30	... 00000000 000
000018E0	30	30	38	34	30	20	30	30	30	30	30	30	34	35	20	30	00840 00000045 0
000018F0	30	30	30	30	30	30	30	30	30	31	34	46	46	37	42	20	000000000014FF7B
00001900	30	30	30	30	30	30	30	20	30	30	30	30	30	30	30	30	00000000 0000000
00001910	30	20	30	30	30	30	30	30	30	30	20	30	30	30	30	30	0 00000000 00000
00001920	30	30	30	20	30	30	30	30	30	30	30	20	30	30	30	30	000 00000000 000
00001930	30	30	30	30	20	30	30	30	30	30	30	30	30	20	30	30	00000 00000000 0
00001940	30	30	30	30	30	30	00	00	C6	B1	66	72	65	65	00	00	00000000...£tfree.
00001950	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00001960	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00001970	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00001980	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....

**Figure 5.1: Decoded Text from Hex Data for File Re-compressed in Apple iTunes**

Additionally, tests such as the analysis of the frequency spectrograms of an audio file can be done and its results can be confirmed with the zero-level sample analysis. In figure 6a, there is a frequency spectrogram of an original audio file recorded on a Tascam DR-05. The frequencies span the range of the audible spectrum up to 20 kHz. In figure 6b, there is a spectrogram of a second-generation file that has a noticeable frequency cutoff above 16 kHz. This is a characteristic of compression and indicative that a file has been re-compressed. If the file that is respective of this spectrogram is found to have zero-level samples at the beginning of the file, it can help as a confirmation that the file has been re-compressed and is not authentic.



**Figure 5.2: Spectrogram of an Original Audio File Recorded on a Tascam-DR05**



**Figure 5.3: Spectrogram of an Audio File that has been Re-compressed Using Freemake**

Audio

## CHAPTER VI

### FUTURE RESEARCH

Additional research that examines how different codecs affect the number of zero-level samples would be helpful in building a database. There are several other audio codecs that have not had testing at this point such as .OGG, ALAC, FLAC, AC3, etc. While the research from this study shows that some audio programs add a linear number of zero-level samples, that is not always the case. Combined with other testing methods such as looking at the meta data of the audio file, this data could assist in determining the authenticity of an audio file. However, the testing could not stand on its own as a means of authenticity. There were some audio programs that did not add any zero-level samples, and for this reason examining the zero-level samples from audio files coming from this program would not yield results.

The study also showed that there was a variance in the number of zero-level samples added based on the device and the original format that it was created in. A proposed test would be to examine the zero-level samples coming from a device that can record audio files in numerous different formats. For example, having the same device record in mono and stereo, recording in different sample rates, and recording in different formats such as .WAV, .MP3, WMA, or others if the device supports numerous different file formats. By testing this, it can be determined what effect different settings or file formats within the same device have on the number of zero-level samples that are added.

## REFERENCES/BIBLIOGRAPHY

<sup>2</sup>Berman, Josh. "ANALYSIS OF ZERO-LEVEL SAMPLE PADDING OF VARIOUS MP3 CODECS." *University of Colorado Denver*, 2013.

Grigoras, Catalin, and Jeff M. Smith. "Forensic Analysis of AAC Encoding on Apple iPhone Voice Memos Recordings." *Audio Engineering Society*, 18 June 2019.

<sup>1</sup>Schroeder, Ernst F., and Johannes Boehm. "Original File Length (OFL) for mp3, mp3PRO and Other Audio Codecs." *Audio Engineering Society*, 22 Mar. 2003.