Global Journal of Researches in Engineering
Electrical and Electronics Engineering
Volume 13 Issue 8 Version 1.0 Year 2013
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Reducing Hearing Aid Power Consumption using Truncated-Matrix Multipliers

By Thomas L. Hemminger & E. George Walters
The Behrend College, United States

Abstract - The traditional platforms for implementing hearing aid algorithms have been application specific integrated circuits (ASIC) and some general purpose DSP chips. One of the most important issues involved in hearing aid design is power consumption, i.e., battery life. This paper introduces an alternative method for implementing hearing aid algorithms by using truncated-matrix multipliers. These designs can offer a significant reduction in power consumption and chip area. However, the approach can often increase computational error but it can be partially compensated for by introducing a method of coefficient shifting of the filter weights. This latter approach significantly reduces the computational error resulting in improved system performance.

Keywords : *truncated-matrix multipliers, hearing aids, power consumption, coefficient shifting, integer processing.*

GJRE-F Classification : *FOR Code: 090699*

Strictly as per the compliance and regulations of :

Reducing Hearing Aid Power Consumption using Truncated -Matrix Multipliers

Thomas L. Hemminger [α] & E. George Walters [σ]

Abstract - The traditional platforms for implementing hearing aid algorithms have been application specific integrated circuits (ASIC) and some general purpose DSP chips. One of the most important issues involved in hearing aid design is power consumption, i.e., battery life. This paper introduces an alternative method for implementing hearing aid algorithms by using truncated-matrix multipliers. These designs can offer a significant reduction in power consumption and chip area. However, the approach can often increase computational error but it can be partially compensated for by introducing a method of coefficient shifting of the filter weights. This latter approach significantly reduces the computational error resulting in improved system performance.

Keywords : truncated-matrix multipliers, hearing aids, power consumption, coefficient shifting, integer processing.

I. Introduction

Most modern hearing aids employ DSP algorithms running on application specific integrated circuits (ASICs) or on modern DSP chips. These algorithms are designed not only to amplify the overall audio signal but to selectively amplify those signals within specific frequency bands. Most all persons suffering from hearing loss lose the upper frequency range of hearing, requiring the audio signal to be separated into specific bands prior to processing [R. Chamberlain et al., 2003], [Y. Wei and Y. Lian, 2006]. For this reason the audio signal is usually separated into a large number distinct bands, or octaves, each amplified with a specific gain, and then the signals are recombined. A compressor stage is often employed to force the final signal to within the hearing range of the user. With the need for extensive signal processing and with the desire to have small unobtrusive devices, one of the main problems with hearing aids is battery life. Many of these devices run on a 1.3V battery drawing less than 2 mA and have a battery lifespan of about 100 hours of normal use [B. Edwards, 1998]. With this in mind we have endeavored to employ truncated-matrix multipliers to reduce the number of components, thus reducing the power consumption. This paradigm also has the added advantage of having less delay than full multipliers which can be beneficial to the user. As stated above, the cost is lower numerical accuracy, but experiment has shown this not to be a significant issue in this work, the reason being that a small increase in truncation noise is beyond most users hearing range. Many current computational methods are based on weighted overlap-add (WOLA) filter banks, windowed finite impulse response (FIR) filter banks, lattice wave digital filter banks (LWDFB), or DFT methods [R. Vicen-Bueno, et al., 2007], [W. Wei, and D. Liu, 2011]. Here it was decided to simulate a hearing aid by employing the windowed FIR method using a Hamming window on the individual frequency bands. The results from using a full multiplier will be compared with those of the truncated-matrix multiplier. This will enable the development of a rough estimate of the power requirements based on the number of components. This paper is organized as follows. Section II describes the truncated-matrix multiplier, and includes the fundamental design and also a method to provide constant correction, to reduce final numerical error. It also provides the rationale for coefficient shifting thus improving the overall accuracy of the result. Section III introduces the simulations that were employed and section IV presents the results. The conclusions and some thoughts for further work are contained in section V.

Author α : Professor of Electrical and Computer Engineering, Penn State Erie – The Behrend College. E-mail : hemm@psu.edu
Author σ : Assistant Professor of Electrical and Computer Engineering, Penn State Erie – The Behrend College. E-mail : egw100@psu.edu

II. Truncated-Matrix Multipliers

Truncated-matrix multipliers are designed by removing several of the least significant columns of the partial product, i.e., these products are not formed [E. G. Walters III and M. Schulte, 2011], [E. G. Walters III and M. Schulte, 2010], [E. G. Walters III, 2012], [T. Erdogan, et al., 2004], [E. E. Swartzlander, Jr., 1999]. As a result, they consume less power, less area, and can have a lower time delay than conventional multipliers. This does come at a cost of less accuracy which may or may not be an issue in certain applications. For example, audio processing mainly concerns perceived sound quality rather than absolutely precise numerical results. Research has shown that video processing does not often need to be precise as a first step in identifying objects in an image, e.g., facial recognition and video surveillance [T. T. Zin, et al., 2011]. In fact, a multi-level approach can be employed whereas the first level of numerical accuracy is lower, but as subjects are narrowed down, the analysis becomes more precise [J. S. Kim, et al., 2011].

In the numerically intensive domain of digital signal processing employment of the truncated

multipliers can provide significant power savings over their full-width counterparts [J. M. Jou, et al., 1999]. These can be direct replacements for standard multipliers with little degradation in numerical performance. In general, FIR filters can have a significant number of smaller floating point coefficients. After converting them to signed integers the result is often a set of coefficients with many leading zeros (positive) or ones (negative) for sign extension. For this reason it is necessary to shift these coefficients to the left prior to multiplication to obtain greater accuracy. However, the operation is only performed on the filter coefficients and not on the incoming data since the bits corresponding to the filters can be modified prior to implementation. This leaves only one set of right shifts when the system is in real-time operation. FIR filters require a very simple set of multiply and add operations as shown in (1) for a T tap filter.

$$y[i] = \sum_{k=0}^{T-1} h[k]x[i-k] \quad (1)$$

Where *x[i]* is the i^{th} value of the input stream and *h[k]* is the set of filter coefficients. When using an odd number of taps the coefficients are symmetric and they yield a linear phase response, which is an attractive quality in audio signal processing. One way to reduce the number of multiplications is to add the two input data values of x_{i-k} and x_{i+k} prior to multiplication by the appropriate filter coefficient but this only increases the complexity of the basic circuit components. Table I shows the coefficients for one of the 63-tap filters used in this work.

The bandwidth of this filter ranges from 500 to 1000Hz and employs a Hamming window. The original rounded integer values of the filter are headed by *h[k]*. The number of left shifts is headed with *S*, and the new left-shifted values are in the next column to the right. The coefficients were developed using MATLAB and then quantized to 16-bit signed integers ranging from -32768 to +32767. Normally, when converting to 16-bit signed integers the coefficients need to be within the range of $[-1, 1-2^{-15}]$ and the multiplier becomes 2^{15}. This is followed by rounding the results. However, here the original coefficients for all the filters had magnitudes within the range of $[-0.5, 0.5-2^{-16}]$ so a multiplier of 2^{16} yielded results within the proper signed integer range, thus eliminating the need to normalize the data. For example, the value of tap *h[21]* in Table 1 was originally -0.049036 which was then multiplied by 2^{16} and rounded to be represented as the 16-bit signed integer -3214. This indicates that the decimal point is implied to be to the right of the most significant (sign) bit. In fact, it is not a good idea to normalize the filters because their relative gains become corrupted by the normalization process. This in turn, unnecessarily complicates computation of the new filter gains so it was decided not to perform that operation. As stated earlier, in order to preserve accuracy it is necessary to shift the bits of each coefficient as far to the left as possible. For example, in the top row the value of *h[0]* is -12 which is shifted to the left by $S = 11$ bits yielding the rightmost column value of -24576. Note that the results in the right column range from -32768 to +32767 thus preserving the sign bit. After multiplication by the corresponding input data point and truncated by the *r* least significant columns the result from each tap is right shifted by the value of S to reestablish the proper magnitude. The result is then added to the summation. It is important to keep in mind that in practice the *r* least significant partial products are not formed in the first place to reduce power consumption. The design is illustrated in Fig. 1 where for simplicity an 8x8 multiplier has been synthesized. Those partial products in the *r* rightmost columns are never formed and there is no corresponding hardware for them.

Table 1: FIR filter with 63 taps

k	h[k]	S	2^S h[k]	k	h[k]	S	2^S h[k]
0	-12	11	-24576	32	4708	2	18832
1	-22	10	-22528	33	4042	3	32336
2	-29	10	-29696	34	3023	3	24184
3	-27	10	-27648	35	1773	4	28368
4	-10	11	-20480	36	438	6	28032
5	29	10	29696	37	-834	5	-26688
6	98	8	25088	38	-1911	4	-30576
7	200	7	25600	39	-2694	3	-21552
8	333	6	21312	40	-3131	3	-25048
9	482	6	30848	41	-3214	3	-25712
10	624	5	19968	42	-2981	3	-23848
11	724	5	23168	43	-2504	3	-20032
12	745	5	23840	44	-1876	4	-30016
13	647	5	20704	45	-1194	4	-19104
14	402	6	25728	46	-546	5	-17472
15	0	15	0	47	0	15	0
16	-546	5	-17472	48	402	6	25728
17	-1194	4	-19104	49	647	5	20704
18	-1876	4	-30016	50	745	5	23840
19	-2504	3	-20032	51	724	5	23168
20	-2981	3	-23848	52	624	5	19968
21	-3214	3	-25712	53	482	6	30848
22	-3131	3	-25048	54	333	6	21312
23	-2694	3	-21552	55	200	7	25600
24	-1911	4	-30576	56	98	8	25088
25	-834	5	-26688	57	29	10	29696
26	438	6	28032	58	-10	11	-20480
27	1773	4	28368	59	-27	10	-27648
28	3023	3	24184	60	-29	10	-29696
29	4042	3	32336	61	-22	10	-22528
30	4708	2	18832	62	-12	11	-24576
31	4939	2	19756				

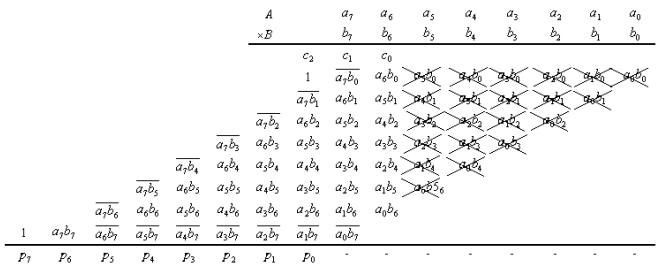

Figure 1 : An 8x8 truncated-matrix multiplier. This employs constant correction with r =6 and k =2

The remaining partial products are added column-wise to produce the desired product. Note that after the addition operation the *k* least significant bits are also truncated so that an 8 bit result is maintained $(p_0 - p_7)$. Of course, the issue here is how to reduce the error from these operations. A number of methods are available in the literature as in [Y. C. Lim, 1992], [L. D. Van and C. C. Yang, 2005] but it was decided to choose a method that has worked well in previous simulations [M. J. Schulte, et al., 1993]. Here, each bit of the multiplier and multiplicand are considered to have equal probability of either being zero or one. In this case their partial product $a_i b_j$ should have an expected value of ¼ so the expected values of the unformed partial products are added to the expected round off error of the product. The sum is then rounded to the least significant column that has been formed. This produces the correction constant *C* which is expressed below in (2).

$$C = round\left(2^{-r}\left(2^{r+k-1} - 2^{r-1} - E_{r\_mean}\right)\right)2^{r} \qquad (2)$$

This value is added to the partial product matrix (see Fig. 1) as bits $c_2\, c_1\, c_0$. The leading ones in some of the rows and the nand operations on some of the elements are necessary to produce the proper signed result. Once the truncated product has been formed it is necessary to compensate for the previous shifting operation on the filter coefficients. Without this procedure the accuracy of the result suffers as described in Fig. 2(a). The correction factor was not introduced here to simplify the figure. In this case the number *B* has several leading zeros and if *r* has a large value, where *r* is the number of truncated columns, the error becomes significant. If the number is small and negative then the most significant bit is one and several of the next most significant bits are also equal to one due to sign extension. As shown in Fig. 2(b) the number *B* has been left-shifted where the shift amount *S* is the number of consecutive bits immediately to the right of the sign bit that have the same value as the sign bit. From this example it is seen that for $B = 0000 b_3 b_2 b_1 b_0$ it should be left shifted by 3 bits to preserve the sign bit. The result is $B = 0 b_3 b_2 b_1 b_0 000$, but note that three zero bits are shifted in from the right, meaning that they reduce the effects of the unformed products. If instead the number is negative with $B = 1111 b_3 b_2 b_1 b_0$ the result from shifting would be $B = 1 b_3 b_2 b_1 b_0 000$ which would also reduce the effects from the unformed products. Shifting to the right by *S* bits after multiplication reduces the error by a factor of 2^S. This does, however, introduce a non-symmetric round-off error. The shifted sum is rounded prior to truncation so this error has a mean value that is close to zero. Rather than adding a one to the right of the least significant bit p_0 prior to truncation, the rounding bit is added to the appropriate column of the partial product matrix prior to shifting by *S* bits. These bits are shown in bold in Fig. 2(b). Here the value of *B* is shifted three places so, $S = 3$, $s_1 = 1$ and $s_0 = 1$. This adds a value of one to the column containing $\overline{a_7 b_0}, a_6 b_1, a_5 b_2,$ and $a_4 b_3$. In this case the multiplier supports shifts from 0 to 3 but if there is no shift $s_1 = 0$ and $s_0 = 0$ and there are no additional bits required for rounding. The number of shifts for each filter coefficient is shown in table 1 where it can be seen that the errors from each multiplication are reduced by a factor of 2^S once the shifting operation has been completed. An n-bit barrel shifter can be used to shift the result back to the appropriate magnitude from the output of the multiplier. In fact, a four-stage barrel shifter can shift from zero to 15 places which cuts down on the amount of required hardware. This aspect is explained in greater detail in [M. R. Pillmeier, et al., 2002].

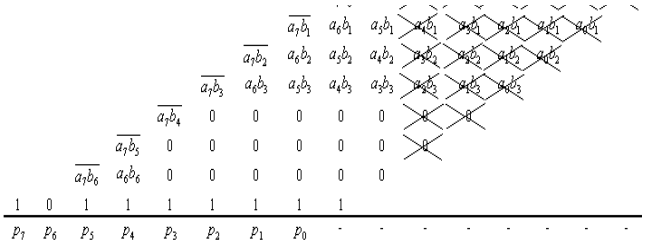

Figure 2 (a): Multiplication without coefficient shifting

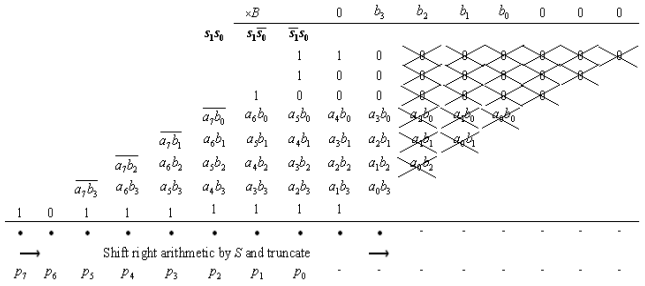

Figure 2 (b): Multiplication with coefficient shifting. $S = 3$

III. Simulations

This section describes the simulations that were employed when evaluating the performance of the multipliers. Fig. 3(a) shows an audiogram from a test subject indicating substantial high frequency loss in the left ear as compared to the right ear (see Fig. 3(b)). From the figure one can see that above a frequency of 2 kHz the subject has significant hearing loss, but at lower frequencies the response is relatively flat. This explains why the subject has little difficulty hearing a voice from a telephone with the left ear since that system is band-limited to about 3 kHz. There are a variety of hearing aid protocols, some having as many as 16 channels or more. However, for this work it was decided that to prove the efficacy of the design a reduced system with five channels would be sufficient. From Fig. 4 it can be seen that five channels were employed corresponding to a frequency range of 0 to 4 kHz, each having its associated gain. The subject's hearing is so poor above 4 kHz that it was deemed unnecessary to amplify sounds above that range. To be consistent with several other systems the sample rate was chosen to be 16 kHz using a 16-bit A/D converter. Each channel was amplified with gains that were determined by the losses indicated in the audiogram for the left ear. Studies have shown that using gain factors to cancel the measured losses shown in the audiogram do not produce acceptable results. For example, a hearing loss of 35-dB indicates an attenuation factor of about $10^{35/10} \approx 3160$. Instead, it has been determined through several studies over the years that using the half gain (or even the third gain) rule yields acceptable results. The half gain rule was chosen for these experiments. It involves amplifying the audio signal by one half of the auditory loss measured in dB. For example, if a person has a 35-dB loss within a specific frequency range it is acceptable to amplify the signal by 17.5-dB. This may seem counterintuitive, and certainly 17.5-dB is not anywhere near one half of 35-dB in terms of true gain or attenuation, but it is known that this is a good starting point when determining channel gains. Referring to Fig. 3(a) the gain to compensate for the channel centered on 3 kHz should be $10^{17.5/10} \approx 56$. To reduce the processing overhead and complexity, this value was converted to 64, and being a power of 2 corresponds to a shift operation of 6 bits. Since the rule is an approximation it was deemed that this would be an acceptable estimate. Fig. 4 shows the block diagram of the hearing-aid system used in this work. It illustrates the five channels of FIR filters that were employed along with their associated signal gains. When performing integer operations overflow is a concern, so the individual channels were not multiplied by their respective gains. In Fig. 4 it appears that channel three is multiplied by 2, channel four by 4, and channel five by 64. Instead, channel five remains unchanged and becomes the reference channel in signal strength. Channel four is divided by 16 (right shift 4 bits), and channel three is divided by 32 (right shift 5 bits) with the remaining channels divided by 64. Of course, the final step is to recombine the channel outputs, next the result can be scaled up to provide the necessary overall gain. Fig. 5 illustrates the responses of each filter superimposed on the same frequency scale.

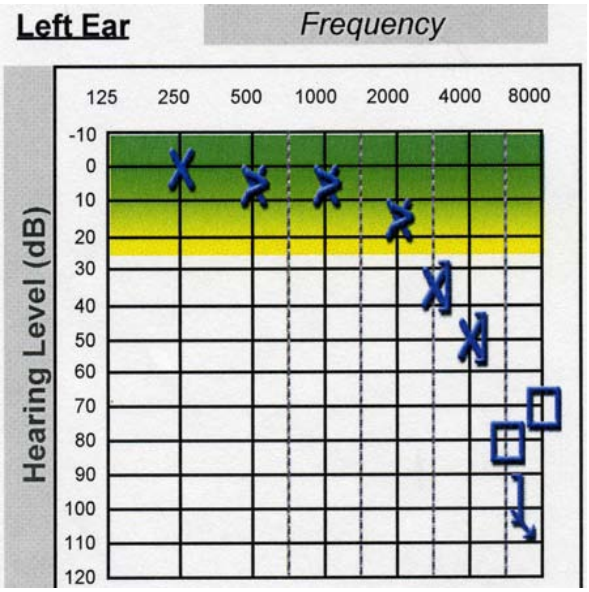

Figure 3 (a): Audiogram from test subject indicating substantial hearing loss in the left ear

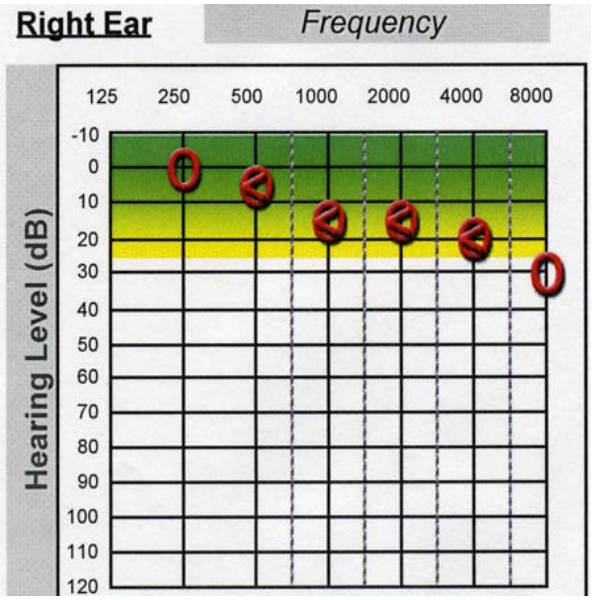

Figure 3 (b) : Audiogram from test subject indicating some hearing loss in the right ear for comparison

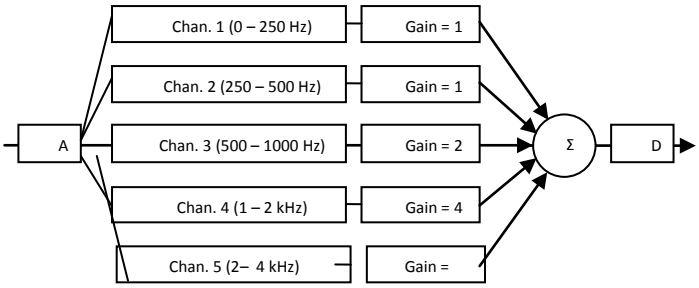

Figure 4 : Block diagram of digital signal processing Stage

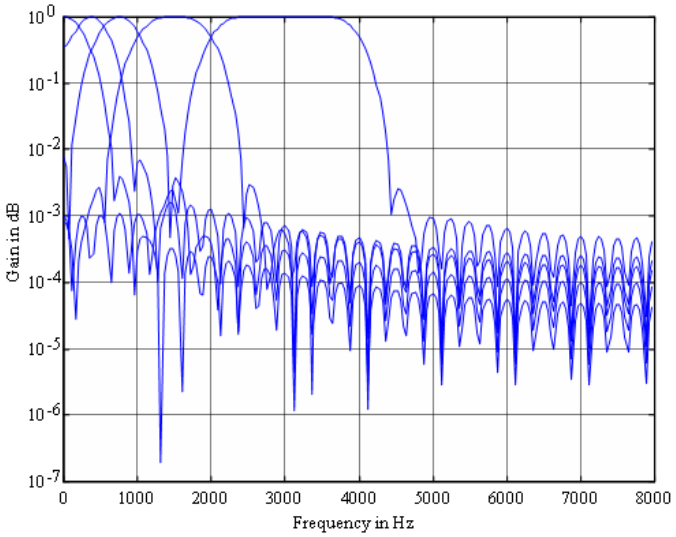

Figure 5 : The five channel filter prior to adjusting the gains

IV. Results

The gains for this project were chosen to compensate for the hearing loss of the subject but also demonstrate a potentially significant dynamic range. It is not necessary for the gains (or attenuations) to be powers of two in order to capitalize on right and left shifts. By employing shifts accompanied by additions or subtractions, effective multiplication can be accomplished with many more gain factors. With 16-bit A/D sampling the quantization noise level is about 96-dB below moderate background levels. Therefore, this aspect will not be an issue since the subjects usually have limited aural acuity and cannot hear beyond a certain range. As a first experiment five sinusoids of equal magnitude were generated at a 16 kHz sample rate and combined into one file. These signals were chosen to correspond to the center frequencies of each filter, e.g., 125 Hz, 375 Hz, etc. The data file was processed with the five filters shown in Fig. 5 using full-width integer multipliers and compared against truncated-matrix multipliers ranging from $r=0$ to $r=15$ The normalized power spectrum of the result was computed as can be seen in Fig. 6 and it is apparent that the signals have been modified by the appropriate gains corresponding to the filter channels. But more importantly this is for $r=15$. The plot resulting from the full-width multipliers looked identical so only this one was included.

The second experiment employed the use of uniformly distributed noise as an input signal. This was chosen so that the entire spectrum would be represented. From both experiments it was found that the error from increasing the value of r was virtually identical. Fig. 7 illustrates the mean-squared error between using full-width multipliers and progressively employing truncated arithmetic on both data sets. These numbers range from -32768 to $+32767$ yet the error for $r=15$ is just over five. Finally, Fig. 8 shows the normalized output spectrum from the white noise input separated into the individual channels, and multiplied by the associated gain factors. This and Fig. 6 illustrate that the weakest part of the subjects aural acuity is compensated for by an appropriate increase in signal gain. Lastly, the output signals from the sinusoids were provided to the test subject. The subject could not distinguish between any of the outputs whether using full-width multipliers or this new paradigm.

Next, it is useful to determine how this design can be beneficial to low-power, miniature devices. In the introduction it was stated that a standard hearing aid consumes about 2 mA and has a battery life of about 100 hours of normal use. It is also commonly known that a great deal of the required power consumption is directly due to the arithmetic units. For a 16x16 bit multiplier the number of AND gates (multipliers) is of order 256. However, with truncation of $r=15$ this could be reduced to $256-[1+2+3+\cdots+15]=136$. This translates into about 53% of the original number. This does not include the barrel shifters but there are far less of those devices. From some earlier work and from the experiments conducted for this paper it appears that there can be an approximate hardware reduction of about 40% compared to conventional methods and it is quite possible that this could translate into roughly a 50 - 60% increase in battery life without any appreciable reduction in signal quality. Even a 40 - 50% increase would be a substantial improvement over the norm translating into about 150 hours of normal use.

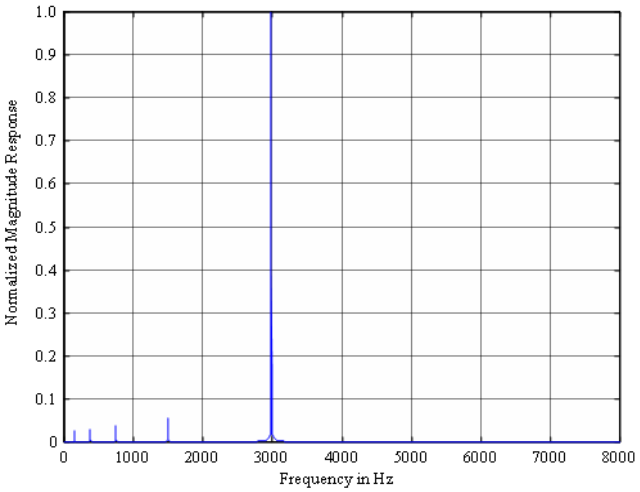

Figure 6 : Responses from five sinusoids. Their amplitudes were originally equal but now reflect the effects of gain

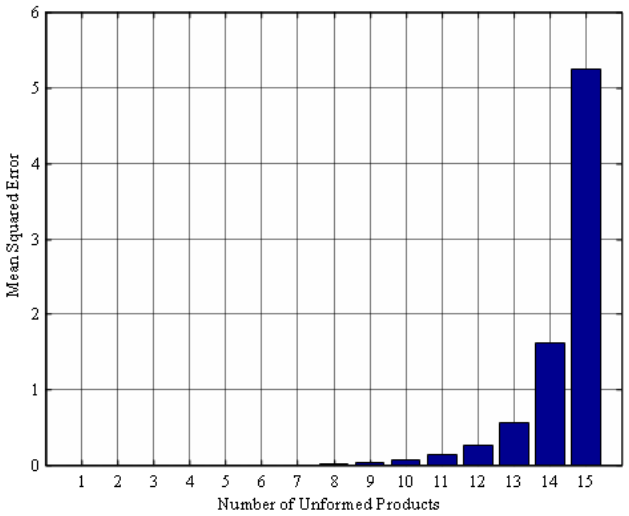

Figure 7 : Error against number of unformed product columns

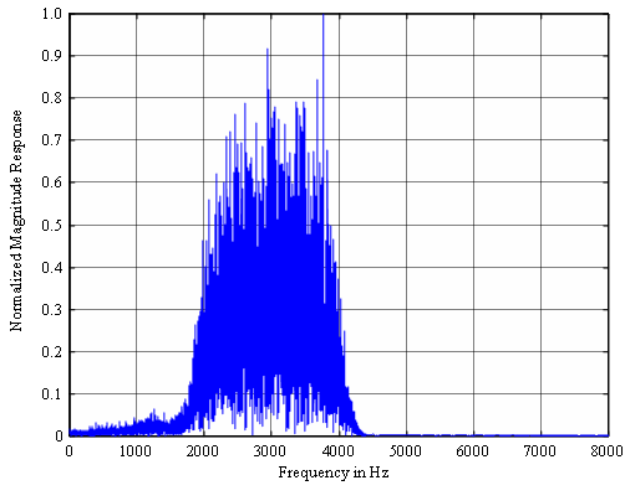

Figure 8 : Response of each filter with its associated gain from a white noise input

V. Conclusion

The results from this work were more encouraging than originally expected. Reducing the number of formed multiplier stages had a small numerical effect that was not discernible in the visual plots. Furthermore, the test subject could not determine the difference between the full-width integer or truncated arithmetic approaches. This was obviously a limited and preliminary experiment and the goal is to place this design on an ASIC so that a full hardware implementation can be realized. The development of high quality signal processing algorithms utilizing low power components is important. It is especially relevant when designing small consumer electronics like cell phones and hearing aids where consumers need to either recharge or replace batteries on a regular basis. There could be a wide application of this technology in the areas of signal and image processing. For example, smart phones, MP3 players, and tablet computers could be designed to employ this technology when performing video and audio processing where data loss is not critical. Other areas that have been suggested are facial and voice recognition along with data reduction techniques, e.g., JPEG and MPEG. For facial recognition the original data can be reduced in resolution using lower numerical accuracy prior to using higher precision methods. Lastly this technology could be employed to develop faster FFT algorithms which could also be useful in a large number of signal processing applications [R. Jiang, 2007].

References Références Referencias

1. R. Chamberlain, J. Goldstein, and D. Ivanovich, "Implementation of Hearing Aid Signal Processing Algorithms on the TI DHP-100 Platform," in *Proceedings of the 37th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, vol. 1, pp. 404-409, November 2003.
2. B. Edwards, "Signal Processing Techniques for a DSP Hearing Aid," in *Proceedings of Circuits and Systems*, Monterey, CA, vol. 6, pp. 586-589, May 31-June 3, 1998.
3. T. Erdogan, E. Zwyssig, and T. Arslan, "Architectural Trade-offs in the Design of Low Power FIR Filtering Cores," *IEE Proceedings. Circuits Devices Systems*, vol. 151, issue. 1, pp. 10-17, 2004.
4. R. Jiang, "An Area-Effiecient FFT Architecture for OFDM Digital Video Broadcasting," *IEEE Trans. Consumer Electronics*, vol. 53, no. 4, pp. 1322-1326, Nov. 2007.
5. J. M. Jou, S. R. Kuang, and R. D. Chen, "Design of Low-Error Fixed-Width Multipliers for DSP Applications," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 46, no. 6, pp. 836-842, June 1999.
6. J. S. Kim, K. H. Yeom, and Y. H. Joo, "Fast and Robust Algorithm of Tracking Multiple Moving

Objects for Intelligent Video Surveillance Systems," *IEEE Trans. Consumer Electronics*, vol. 57, no. 3, pp. 1165-1170, Aug. 2011.
7. Y. C. Lim, "Single-Precision Multiplier with Reduced Circuit Complexity for Signal Processing Applications," *IEEE Transactions on Computers*, vol. 41, no. 10 pp. 1333-1336, October 1992.
8. M. R. Pillmeier, M. J. Schulte, and E. G. Walters III, "Design Alternatives for Barrel Shifters and Rotators," in Proceedings of the SPIE: Advanced Signal Processing Algorithms, Architectures and Implementations XII, vol. 4791, Seattle, WA July 2002, pp. 436-447.
9. M. J. Schulte, E. E. Swartzlander, Jr. "Truncated Multiplication with Correction Constant," *VLSI Signal Processing VI*. Eindhoven, Netherlands: IEEE Press, October 1993, pp. 388-396.
10. E. E. Swartzlander, Jr., "Truncated Multiplication with Approximate Rounding," in *Proceedings of the 33rd Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, vol. 2, pp. 1480-1483, October 1999.
11. L. D. Van and C. C. Yang, "Generalized Low-Error Area-Efficient Fixed-Width Multipliers," *IEEE Transactions on Circuits and Systems-I: Regular Papers*, vol. 52, no. 8, pp. 1608-1619, 2005.
12. R. Vicen-Bueno, R. Gil-Pita, M. Utrilla-Manso, and L. Alvarez-Perez "A Hearing Aid Simulator to Test Adaptive Signal Processing Algorithms," in *IEEE International Symposium on Intelligent Signal Processing,* Alcala de Henares, Spain, pp. 1-6, October 2007.
13. E. G. Walters III, "A Design-Space Exploration Tool for Low-Power DCT and IDCT Hardware Accelerators," in *Proceedings of the IEEE 16th International Symposium on Consumer Electronics*, Harrisburg, PA, pp. 1-5, June 2012.
14. E. G. Walters III and M. Schulte, "Fast, Bit-accurate Simulation of Truncated-matrix Multipliers and Squarers," in *Proceedings of the 44th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, pp. 1139-1143, November 2010.
15. E. G. Walters III and M. Schulte, "Truncated-Matrix Multipliers with Coefficient Shifting," in *Proceedings of the 45th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, pp. 176-180, November 2011.
16. Wei and Y. Lian, "A 16-Band Nonuniform FIR Digital Filterbank for Hearing Aid," in *Proceedings of the IEEE Biomedical Circuits and Systems Conference*, London, UK, pp. 186-189, November 29-December 1, 2006.
17. W. Wei, and D. Liu, "A Design of Digital FIR Filter Banks with Adjustable Subband Distribution for Hearing Aids," in *8th International Conference on Information, Communications, and Signal Processing*, Beijing, China, pp. 1-5, November 2011.
18. T. T. Zin, P. T. Hiromitsu Hama, and T. Toriu, "Unattended Object Intelligent Analyzer for Consumer Video Surveillance," *IEEE Trans. Consumer Electronics*, vol. 57, no. 2, pp. 549-557, May 2011.

This page is intentionally left blank