EUROREGIO
BNAM2022
Joint Acoustics Conference
9th-11th of May, Aalborg, Denmark

# The Effect of Fixed-point Arithmetic on Low Frequency Sound Zone Control

**Peter Koch**[1,*], **Jan Østergaard**[1]

[1]Department of Electronic Systems, Aalborg University, Aalborg, Denmark.

*pk@es.aau.dk

**Abstract**

By use of several loudspeakers it is possible to create constructive and destructive acoustic interference leading to spatial sound zones with high (bright zone) and low (dark zone) sound pressure levels, respectively. The loudspeaker control signals are commonly created by filtering the audio signals using Finite Impulse Response (FIR) control filters. We are interested in using finite precision multiply-accumulate arithmetic in the FIR filter operations in order to reduce the complexity of a potential hardware implementation. In a simulation study based on a setup with 8 loudspeakers and for the frequency range 50 – 600 Hz, we demonstrate that using only 8 and 12 bits signed fixed-point arithmetic for the multiplier and adder, respectively, reduces the mean acoustic contrast ratio between the bright and the dark zones by only 0.1 dB as compared to when using 32 bits arithmetic. Reducing the word length of the adder below 12, significantly increases the sound pressure in the dark zone.

**Keywords**: sound zones, fixed point arithmetic, FIR filtering, variable word length simulation.

## 1 Introduction

Control of personal or individual sound zones refers to a specific problem within sound field control, where one is interested in generating individual listening zones for separate listeners in some enclosed space, for example, a room [1] or a car cabin, [2]. It is common to consider the control of two sound zones in a room, and where the sound field outside the zones is not controlled, [3]. In one of the zones, one would like to reproduce a specific sound signal with a high quality and at a high sound pressure level. This zone is usually referred to as the *bright* zone. In the other sound zone, one would like to reduce the sound pressure level as much as possible so that it is harder to hear the audio signal. This zone is usually referred to as the *dark* zone. By combining a pair of bright and dark zones, it is then possible to achieve different audio content in different spatial locations in the same room.

To control the low frequencies in the sound zones, it is advantageous to make use of knowledge of the acoustical properties of the enclosed space – such as the room transfer functions (RTFs), [4]. For the mid-frequencies, conventional beam forming techniques are often sufficient, whereas for the highest frequencies, the directivity of the loudspeakers can be exploited, [4].

The design of sound zones is currently an active area of research, see for example, [5, 6, 7, 8, 5, 9, 10]. To take into account the RTFs of the enclosed space, it is often necessary to use long FIR filters having several hundreds taps. Moreover, to achieve a high degree of acoustic separation (contrast) between the zones, it is generally necessary to use several loudspeakers. Since the individual loudspeakers requires different filtered signals, the complexity in terms of hardware requirements, execution time, and energy consumption of the filter operations can be quite significant. To reduce this complexity it is possible to use finite word length arithmetic for the implementations of the FIR filter operations, [11].

The aim of this paper is to demonstrate in a simulation study the effect that finite precision arithmetic has on the resulting acoustic separation between the bright and the dark zone as well as on the audio quality in the bright zone. We will use a conventional FIR filtering approach to control the bright as well as the dark zone. We will be using fixed-point implementations of the multiplier and the adder used for the FIR filter operations, and we will focus on the low frequency region from 50 – 600 Hz. The RTFs used in the study to evaluate the performance are real measurements of a room equipped with 8 loudspeakers. Our experiments reveal that in particular it is the suppression of sound in the dark zone that is destroyed when the accuracy of the arithmetic operations becomes too low. We also observe that a high degree of acoustic contrast and sound quality (in a squared error sense) is possible using only 8 bits for the multiplier and 12 bits for the adder in the FIR filtering operations.

## 2 Background on FIR filtering based sound zones

Let us assume the availability of a set of $L > 0$ loudspeakers, which are arbitrarily distributed within a room. We adopt the notation and setup from [10], and assume that the sound pressure is known in $M_b$ and $M_d$ points in the bright and dark zones, respectively. For example, this knowledge can be obtained by placing microphones in the sound zones of the room. The sound pressure $p_b^{(m)}$ at the $m$-th point in the bright region is given by the combined output of all $L$ loudspeakers convolved with the their room impulse responses. Specifically, let $u[k]$ be the $k$-th sample of the single audio signal to be filtered and played out by the loudspeakers. Moreover, let $\bar{w}^{(l)} \in \mathbb{R}^{N_w}, \ell = 1, \dots, L$, be the impulse response of the control filter for the $\ell$-th loudspeaker, where $N_w$ denotes the length in samples of the impulse response. We limit our attention to FIR filters, and $N_w$ is therefore finite. Finally, let $\bar{h}_b^{(m,\ell)} \in \mathbb{R}^{N_h}$ be the room impulse response (RIR) between the $\ell$-th loudspeaker and the $m$-th point in the bright zone, and where we limit the response to $N_h$ samples. We define $\bar{h}_d^{(m,\ell)}$ in a similar manner for the dark zone. Using this notation, we can express the sound pressures $p_b^{(m)}[k]$ and $p_d^{(m)}[k]$ at time $k$ as follows, [10]:

$$p_b^{(m)}[k] = \sum_{\ell=1}^{L} \left( \bar{h}_b^{(m,\ell)} \star \bar{w}^{(\ell)} \star u \right)[k] = \sum_{\ell=1}^{L} \sum_{j=0}^{N_w-1} \sum_{i=0}^{N_h-1} \bar{h}_b^{(m,\ell)}[i] \bar{w}^{(\ell)}[j] u[k-i-j], \tag{1}$$

$$p_d^{(m)}[k] = \sum_{\ell=1}^{L} \left( \bar{h}_d^{(m,\ell)} \star \bar{w}^{(\ell)} \star u \right)[k] = \sum_{\ell=1}^{L} \sum_{j=0}^{N_w-1} \sum_{i=0}^{N_h-1} \bar{h}_d^{(m,\ell)}[i] \bar{w}^{(\ell)}[j] u[k-i-j], \tag{2}$$

where $\star$ denotes linear convolution. We assume the filters and the RIRs to be time invariant.

The average accumulated squared sound pressure level (or sound *energy*) $P_{\text{bright}}$ and $P_{\text{dark}}$ in the bright and dark zones, respectively, are then given by:

$$P_{\text{bright}} \triangleq \frac{1}{N_u M_b} \sum_{m=1}^{M_b} \sum_{k=0}^{N_u-1} |p_b^{(m)}[k]|^2, \quad P_{\text{dark}} \triangleq \frac{1}{N_u M_d} \sum_{m=1}^{M_d} \sum_{k=0}^{N_u-1} |p_d^{(m)}[k]|^2, \tag{3}$$

where $N_u$ denotes the length of the time-domain audio signal $\{u[k]\}$. A similar notation applies for the energy $P_{\text{dark}}$ in the dark zone. We are now in a position to introduce the mean acoustic contrast ratio (expressed in dB) and the normalized mean squared error (MSE), which are defined as follows:

$$C = 10 \log_{10} \left( \frac{P_{\text{bright}}}{P_{\text{dark}}} \right) [\text{dB}], \quad Q = \frac{\sum_{m=1}^{M_b} \sum_{k=0}^{N_u-1} |p_b^{(m)}[k] - \tilde{p}_b^{(m)}[k]|^2}{\sum_{m=1}^{M_b} \sum_{k=0}^{N_u-1} |\tilde{p}_b^{(m)}[k]|^2}. \tag{4}$$

where $\tilde{p}_b^{(m)}[k]$ is a specific desired target pressure level at time $k$ at the $m$-th position in the bright zone. We will be using the contrast ratio and the normalized MSE to quantify the performance of the system, when we are changing the precision of the arithmetic operations involving the convolutions between the control filters and the audio signals. The contrast ratio quantifies the amount of ”separation” between the two zones, and the normalized MSE quantifies the quality of the resulting audio signal in the bright zone.

# 3 Finite word length FIR filtering in sound zones

The convolutions described by Equations (1) and (2) can be separated into two stages. In the first stage, the audio signal $\{u[k]\}$ is convolved with the control filters $\{\bar{w}^{(\ell)}\}$, and in the second stage the *filtered* audio signal is played out and will thereby be convolved with the impulse responses $\{\bar{h}_b^{(m,\ell)}\}$ and $\{\bar{h}_d^{(m,\ell)}\}$ of the room. In this work, we will focus on the first stage, where we will modify the filter coefficients $\{\bar{w}^{(\ell)}\}$ as well as the convolution operator $\star$ in order to model the effect of using finite precision arithmetic. The filtering operations in the second stage are not affected by the modifications in the first stage.

## 3.1. Quantization of filter coefficients

Operating in a finite word length environment where signed arithmetic operations are needed, several potential number representations may be considered, ranging from the simple ”signed magnitude” notation to advanced representations such as redundant binary number systems. The number representation being used highly impacts the implementation cost as well as the execution time of the overall application. For instance, the advantage of a redundant number system is that it can perform addition in a constant time independent of the word length, the drawback being a significant hardware overhead demanded by multiple bits per digit as well as input/output converters needed for interfacing against a traditional binary number representation.

Since our primary aim in this work is to investigate the numerical robustness of sound zones operated in a finite word length environment, our experiments will be conducted using a $d$-bit 2’s complement fixed-point number representation which easily handles signed arithmetic operations in the dynamic range $[-1;1[$, but which at the same time may be far from optimal in terms of implementation cost and execution time. We will address specific hardware implementation issues in a future work.

Preparing the control filters for fixed-point execution, we initially quantize the $N_w$ coefficients of each of the $L$ filters. To do this, we scale all coefficients by the same factor so that all coefficients are within $[-1,1]$. Specifically, let $\bar{w}^{(\ell)} \in \mathbb{R}^{N_w}$ denote the impulse response of the $\ell$-th filter. Then let $c_\ell = \max_{\ell,j} |\bar{w}^{(\ell)}[j]|$, where $\bar{w}^{(\ell)}[j]$ denotes the $j$-th element of the vector $\bar{w}^{(\ell)}$, and $|\cdot|$ denotes the absolute value operator. We form the normalized filters $\bar{w}^{(\ell)} c_\ell^{-1}$, and next quantize these filters to word length $d > 0$ using the following operations:

$$\hat{w}^{(\ell)}[j] = \left\lfloor \frac{\bar{w}^{(\ell)}[j]}{c_\ell} 2^{d-1}(1-\epsilon) \right\rfloor 2^{-(d-1)} \quad j = 0, \dots, N_w - 1, \tag{5}$$

for $\ell = 1, \dots, L$, where $\lfloor \cdot \rfloor$ denotes rounding towards the nearest integer from below, e.g., $\lfloor -0.6 \rfloor = -1$, and $\lfloor 0.6 \rfloor = 0$. Multiplying by $(1-\epsilon)$, where $0 < \epsilon \ll 1$ is a small positive constant, guarantees that all the filter coefficients are in the range $[-2^{d-1}, 2^{d-1} - 1]$ before being quantized to nearest integer. For the special case where $d = 0$, we simply replace the filters by a unit impulse, i.e., $\hat{w}^{(\ell)} = [1, 0, \dots, 0]^T, \forall \ell$. Thus, in this case we do not control the sound field in the sound zones but simply play out the audio without any filtering taking place (except the convolutions of the audio with the RIRs).

## 3.2. 2’s complement fixed-point arithmetic

In order to conduct the control-filter computations, i.e., multiply-accumulate operations reflecting as accurately as possible a real-time hardware execution, we initially design bit-true 2’s complement Matlab-based multiplier and adder simulation models. We therefore briefly introduce the underlying mathematical fixed-point operations applied in these models which are next implemented with appropriate input/output converters such that their input operands and the resulting product/sum can be represented as floating point numbers, [12].

**Radix-4 multiplication** Given two $d$-bit numbers $X$ and $Y$ being the multiplicand and the multiplier, respectively. Expressing $Y$ as a 2's complement number, which represents the individual filter coefficients $\hat{w}^{(\ell)}[j]$, we use a notation where the Most Significant Bit (MSB) is indexed as $0$. Normally, the MSB is indexed $d-1$, but in this FIR filter context where we scale the input signal ($X$) and the coefficients ($Y$) to the dynamic range $[-1;1[$, the 0-indexing of MSB is a notation which conveniently is used to express $Y$ in binary notation, and thus the product $P$ as:

$$Y = -y_0 + \sum_{j=1}^{d-1} y_j \cdot 2^{-j}, \quad P = Y \cdot X = -y_0 \cdot X + \sum_{j=1}^{d-1} (y_j \cdot X) \cdot 2^{-j}, \tag{6}$$

where the fixed point is located immediately to the right of the sign bit $y_0$.

From Equation (6) we see that the product consist of $d$ partial products which are individually left-shifted and added, starting from the Least Significant Bit (LSB) end. In order for this to work, appropriate sign-extension has to be enforced prior to the addition. Since $X$ and $Y$ both have format Q1.$d$-1, the product is format Q2.2$d$-2 which due to two identical sign bits (in case of no overflow) is easily adjusted into format Q1.2$d$-1 by a logical left shift. Now, using the 2's complement notation and the assumption that $d$ is an even number (identical arguments can be derived for $d$ odd), the multiplier is parted into two sums represented by the odd and the even indices, respectively;

$$Y = -y_0 + \sum_{j=1,odd}^{d-1} y_j \cdot 2^{-j} + \sum_{j=2,even}^{d-2} y_j \cdot 2^{-j} \tag{7}$$

Adding and subtracting the "even indexed sum" on the right-hand side of Equation (7), it can be rewritten as

$$Y = -y_0 + \sum_{j=1,odd}^{d-1} y_j \cdot 2^{-j} + \sum_{j=2,even}^{d-2} y_j \cdot 2^{-j+1} - 2 \cdot \sum_{j=2,even}^{d-2} y_j \cdot 2^{-j-1}. \tag{8}$$

If $Y$ is appended at the LSB-end with a bit $y_d$, which is identically equal to 0 and therefore does not alter the numerical value of $Y$, the two "even indexed sums" can now be reformulated in terms of two identical but "odd indexed sums";

$$\sum_{j=2,even}^{d-2} y_j \cdot 2^{-j+1} = \sum_{j=1,odd}^{d-1} y_{j+1} \cdot 2^{-j} \quad \text{and} \quad -2 \cdot \sum_{j=2,even}^{d-2} y_j \cdot 2^{-j-1} = -2 \cdot \sum_{j=1,odd}^{d-1} y_{j-1} \cdot 2^{-j} + y_0. \tag{9}$$

The multiplier $Y$, and hence the product $P$ can therefore be written as:

$$Y = \sum_{j=1,odd}^{d-1} (y_j + y_{j+1} - 2 \cdot y_{j-1}) \cdot 2^{-j}, \quad P = \sum_{j=1,odd}^{d-1} (z_j \cdot X) \cdot 2^{-j} \tag{10}$$

where

$$z_j = y_j + y_{j+1} - 2 \cdot y_{j-1}; z_j \in \{0, \pm 1, \pm 2\}. \tag{11}$$

From Equations (10) and (11) it is concluded that $P$ is the sum of $d/2$ left-shifted and sign-extended partial products (PP) which can take on the values $\{0, \pm X, \pm 2X\}$ depending on the bit pattern of three consecutive bits of the multiplier $Y$, starting with $y_0$ at the MSB-end.

In the general case, where $d$ can be even or odd, we obtain:

$$Y = \sum_{j=0}^{\lceil \frac{d}{2} \rceil - 1} (y_{2j+1} + y_{2j+2} - 2 \cdot y_{2j}) \cdot 2^{-(2j+1)}, \quad P = \frac{1}{2} \cdot \sum_{j=0}^{\lceil \frac{d}{2} \rceil - 1} X \cdot z_j \cdot 4^{-j}. \tag{12}$$

From Equation (12) it is easily noticed that the PPs are individually shifted two bit positions against each other, i.e., Radix 4, and similarly that the final product is obtained only after a 1-bit right shift of the sum of the $\lceil \frac{d}{2} \rceil$ PPs which are all represented as sign-extended 2's complement numbers. In our model, the PPs are calculated sequentially (despite that a parallel computation is possible in a dedicated hardware configuration) and next converted into a floating point representation before they are consecutively added. Since the PPs initially are represented as fixed-point numbers, conducting the additions using floating point arithmetic significantly simplifies the simulation model but does not alter the accuracy, i.e., the resulting floating point product is generated with an accuracy equivalent to a 2's complement number in Q$1.2d$-1 format. The word length $d$ is an adjustable parameter in our model enabling experimentation with varying accuracy of the calculated products.

**2's complement addition** One of the most unique features of 2's complement numbers, as compared to other more straight forward number representations like for instance the signed magnitude representation, is the possibility to perform signed addition (and thus also subtraction) directly on the two input operands. Due to this, contrary to multiplication, there is no need to distinguish between algorithms for signed and unsigned addition.

Consequently, a traditional $d$-bit Ripple Carry Adder (RCA) or a Carry Look-ahead Adder (CLA), eventually extended with overflow detection, can be opted for in a signed signal processing context like the one addressed in this work. While both of these two adder concepts perform a numerically exact computation (for a given word length $d$), the CLA introduces a mechanism which in a parallel manner pre-calculates the carry at several selected bit positions throughout the total word length, thus reducing the worst case propagation delay, i.e., the overall execution time, as compared to an RCA of same word length. The overhead however, being a significantly higher complexity as well as an irregular circuit layout.

Since in this work we want to prepare for the least complex hardware topology, the obvious choice for a 2's complement adder simulation model therefore is the $d$-bit RCA which generates the sum of two Q$1.d$-1 numbers by performing bit-wise iterative addition of the operands, starting from the LSB-end. At the same time, the RCA performs addition of carry information from the lower consecutive bit position, using a 3-2 adder compressor at each bit position. In this work, the carry into the LSB is defined identically equal to 0.

Our RCA-based simulation model takes as input two floating point operands which in the FIR filter context are the individual filter tap products and the accumulated sum, respectively. Using floating point number representation for the inputs makes it easy to interface against the products generated by the multiplier. Furthermore, from a numerical perspective the product accumulation can be conducted in any precision, i.e., single-, double- or overflow precision, since internally our model implements a $d$-bit RCA where the word length is an adjustable parameter which allows experimentation with varying addition accuracy.

For all FIR control filters we conduct appropriate numerical scaling of the input signal in order to avoid an overflow condition at the output variable, and similarly we perform online check for any overflow internally in the structure. For this reason no guard bits are needed in our adder model. The adder thus produces a sum which initially is derived in Q$1.d$-1 format, and next converted to and presented at the output as a floating point number with an identical accuracy.

## 4 Simulation study

In this section, we consider an experimental setup having $L = 8$ loudspeakers and two sound zones, a bright and a dark zone. We let the audio signal $\{u[k]\}$ be a low-frequency 50 – 600 Hz band-limited white Gaussian noise signal sampled at 1.2 kHz. The audio signal is filtered by the $L$ FIR control filters $\{\bar{w}^{(\ell)}\}, \ell = 1, \ldots, L$, before being played out. To design the control filters, we use the design method presented in [13]. We use real measured RIRs $\{\bar{h}_b^{(m,\ell)}\}$ and $\{\bar{h}_d^{(m,\ell)}\}$ for the design of the control filters and when simulating the resulting performance. The room was of size 7.00x8.12x3.00 meters. Each control filter has length $N_w = 100$. An example of one of the filter impulse responses is shown in Figure (1) (left), and similarly an example of one the

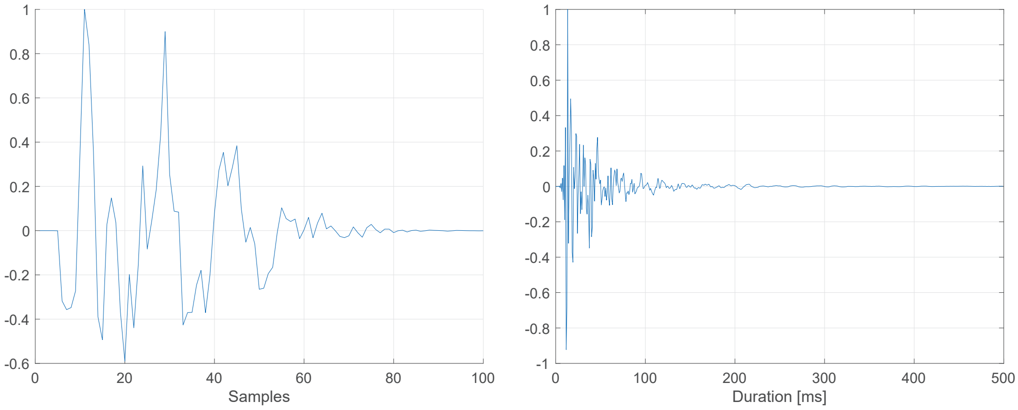

Figure 1: Left: Example of one of the control filters. Right: Example of one of the RIRs used when evaluating the resulting performance of the sound zones.

Table 1: Contrast ratio in dB as a function of the word length of the adder, and for fixed 8 and 16 bits multipliers. The contrast ratio when using 32 bits adder and multiplier is 23.09 dB.

| Multiplier / Adder | 6 bits | 8 bits | 10 bits | 12 bits | 14 bits |
|---|---|---|---|---|---|
| 16 bits | 15.08 dB | 20.56 dB | 22.72 dB | 23.06 dB | 23.09 dB |
| 8 bits | 14.95 dB | 20.67 dB | 22.72 dB | 22.98 dB | 22.98 dB |

RIRs is shown in Figure (1) (right).

For reference purposes, we initially calculate $i)$ the normalized MSE, and $ii)$ the contrast ratio using a 32 bit word length, both for the multiplier and for the adder in the control filters. We obtain reference values equal to -9.89 dB and 23.09 dB, respectively, for the two performance metrics.

We next demonstrate the effect of replacing the 32-bit arithmetic operations by shorter word length multiplications and additions. Figure (2) (left) shows the normalized MSE as a function of the adder word length for different multiplier word lengths, and similarly Figure (2) (right) shows the corresponding acoustic contrast ratios. Table 1 shows the contrast ratio as a function of the adder word length, and for a fixed 8 and 16 bits multiplier.

The results shown in Figure (2) represents an average performance covering all frequencies. To better illustrate the impact of finite-precision arithmetic on the resulting sound zones, we therefore introduce Figures (3) and (4) which show the resulting Power Spectral Densities (PSD) for the bright and dark zones, respectively.

These experiments illustrate several interesting performance features of sound zones operated in a reduced numerical accuracy environment. First and foremost we observe a significant difference among the bright and the dark zone. The bright zone is mostly unaffected by a reduction in the multiplication accuracy given a fixed reference adder word length, Figure (3) (right), the exception being the top-most 50 Hz of the frequency range where a 4-bit multiplier word length leads to an approximately 15 dB degradation of the sound field as compared to longer multiplier word lengths.

Maintaining the reference multiplier word length for a reduced adder accuracy, we observe a somewhat more sensitive bright zone in the upper 100 Hz of the investigated frequency range when the adder word length is decreased below 10 bits, Figure (3) (left). Particularly, despite that the adder word length seems to have no or very little impact in the center part of the frequency band, an up to 20 dB degradation is discovered at the band edges when applying a 4-bit adder.

For both of the above discussed situations we explain the increased sensitivity in the upper part of the frequency band by the following considerations. All control filters are implemented as ordinary transversal filters, i.e.,

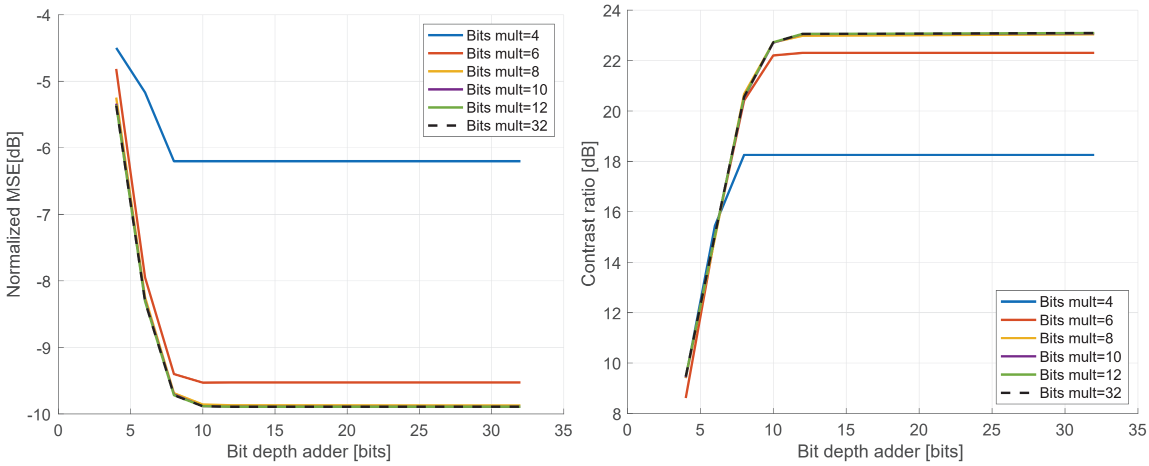

Figure 2: Left: Normalized MSE as a function of the adder word length for given multiplier word lengths. Right: Acoustic Contrast Ratio as a function of the adder word length for given multiplier word lengths.

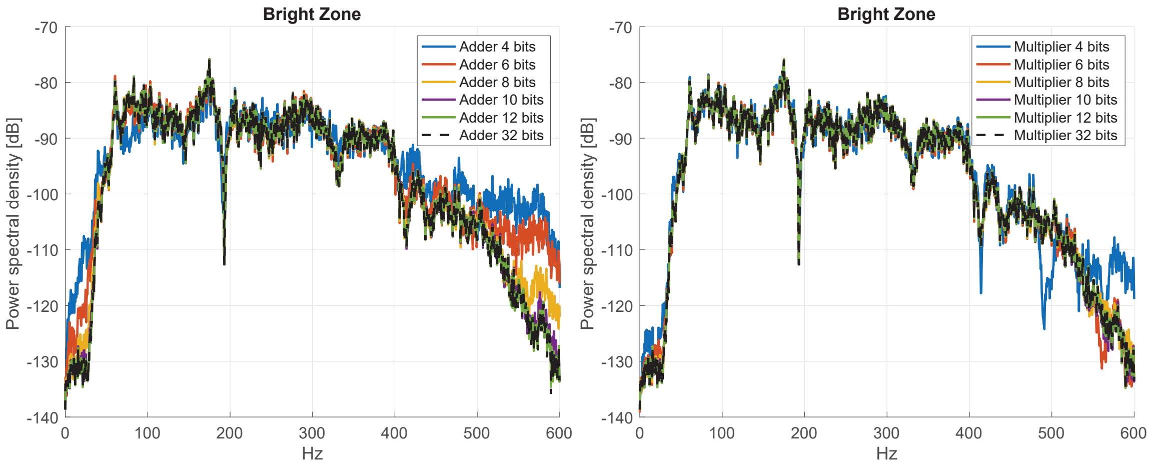

Figure 3: Power spectral density of the bright zone. Left: 32-bit reference multiplier and shorter word length adder. Right: 32 bit reference adder and shorter word length multiplier.

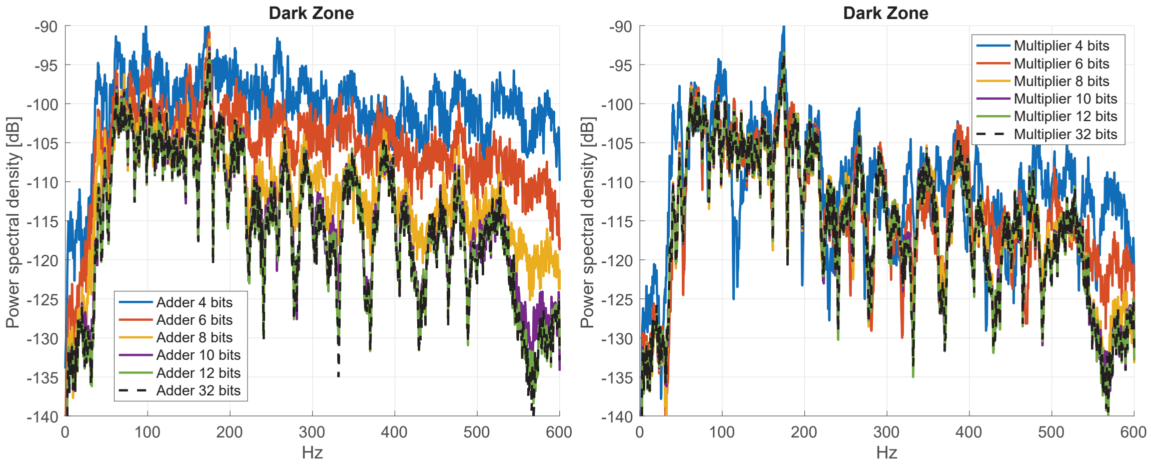

Figure 4: Power spectral density of the dark zone. Left: 32-bit reference multiplier and shorter word length adder. Right: 32 bit reference adder and shorter word length multiplier.

the tapped delay line is storing the input samples $u[k-j]$, $j = 0, ..., N_w - 1$, represented as 2's complement numbers, all scaled to comply with the dynamic range $[-1; 1[$. If the frequency of the input signal is increased (for a fixed sample frequency), the probability for changing the MSB part of the word, which is a representation of the sign, is similarly increased among consecutive input samples. Consequently, for a bit $j$ in the MSB part of the word, the temporal correlation among consecutive samples, given as

$$\rho_j \propto E[u_j[k]u_j[k-1]] \tag{13}$$

therefore decreases, which indicates an increased transition probability, [14].

After all $N_w$ samples in the delay line are multiplied with their corresponding constant filter coefficients, the individual products from the FIR taps similarly show decreased temporal correlation for the bits at the MSB end when the input signal frequency becomes higher. When the word length in the adder chain is reduced, a proportionally larger part of the operand bits are used for sign information, and thus fewer bits are available to represent fractional accuracy. In a "high frequency" scenario, where the sign information becomes increasingly more important, the accuracy of the product summation therefore suffers from proportionally fewer fractional bits, thus impacting the overall numerical quality of the sum. This set of arguments also explains why the sound zone application is less sensitive towards a reduction in the multiplier word length as compared to the adder word length.

Concerning the dark zone, we generally found a much more pronounced dependency of the arithmetic accuracy. We explain this mainly by the fact that creating a dark zone is significantly more complicated as compared to generating a bright zone. In the bright zone, the constructive interference increases the sound pressure level by a certain factor depending upon the number of sources. On the other hand, in the dark zone, ideally there is zero sound pressure due to a complete cancellation of all the direct and reflected sound paths from the $L$ loudspeakers. This is of course not possible in practice and our simulations also illustrate that the sound pressure level is not zero in the dark zone. For the dark zone scenario we therefore also discover exactly the same phenomenons in the "high frequency" regions as discussed above for the bright zone. Moreover, there are several other important observations associated with the dark zone.

Consider first the fixed reference adder scenario, Figure (4) (right). Here we observe an interesting behaviour, namely a somewhat inconsistent PSD relation between the reference multiplier and the shorter word length multipliers. Normally, one would think that a word length reduction leads to a consistent performance degradation, but for several frequencies in the mid-range region we see the opposite effect, particularly for 6- and 4-bit multipliers. One possible explanation might be that the white Gaussian noise input signal, due to its random nature combined with a word length dependent accuracy reduction in the FIR filter tap products, which next are to be added, enables an enhanced "self cancellation" effect of the direct and reflected sound contributions at some arbitrary frequencies – in several cases up to 15 dB better than the reference. Overall however, the PSD obtained by the 32-bit reference and the lower word length multipliers agrees, although with a generally higher fluctuation throughout the entire frequency range as compared to the bright zone scenario.

Additionally, the experiments indicate that significant degradation, in comparison to the 32-bit reference multiplier, only occurs for a multiplier word length less than 8 bit which is therefore considered as the lower bound word length for the multiplier. This observation complies with result also illustrated in Figure (2).

Finally, for the fixed reference multiplier scenario, Figure (4) (left), we observe similar behaviours as already discussed. One exception though being the pronounced full-range performance degradation discovered for a continuous adder word length reduction. Not only does the PSD degrade with up to 30 dB in the high end of the frequency band, but throughout the entire frequency band the degradation, except for a very narrow band around 180 Hz, never falls below 10 dB (using a 4-bit adder). Similarly, we observe a very distinct, almost linear performance degradation when the adder word length is decreased in 2-bits steps. Deviation from the 32-bit reference adder becomes pronounced for a 10-bit and lower word length adder, and similarly more pronounced for increased frequency. These findings clearly indicate that the dark zone is specifically sensitive towards the adder word length which we explain by the same set of arguments as introduced for the bright zone with fixed reference multiplier.

## 5 Conclusions

In order to prepare recently developed Sound Zone Control algorithms for implementation in a real-time fixed-point reconfigurable hardware environment with the possibility for individual word length selection of the arithmetic units – and thus potential power-, time-, and area savings – we investigate the sound zone performance as related to the necessary multiplier- and adder word lengths. Using 2's complement based Ripple Carry Addition and Radix-4 Multiplication, we demonstrate in terms of $i$) Normalized Mean Square Error, $ii$) Acoustic Contrast Ratio, and $iii$) Power Spectral Density, in the bright as well as in the dark sound zone, that the multiplier- and adder word lengths can be reduced to 8 bit and 12 bit, respectively, when compared against a 32-bit reference word length.

For these specific word lengths, we conclude that it is possible to maintain a reduction in the ratio of sound pressure levels between the bright and the dark zones of at most 0.1 dB which is considered sufficiently small in order not to disturb the overall sound perception individually in the two zones.

Furthermore, our studies have clearly demonstrated that the dark zone is difficult to construct when being operated in a reduced word length scenario. Maintaining a destructive interference, in a given acoustic/physical environment, requires a certain amount of arithmetic operations, i.e., a necessary order of the control filters being executed with a sufficient numerical accuracy. We demonstrate that in particular a reduced adder word length impacts the possibility to maintain a high fidelity dark zone, which on the other hand is significantly less sensitive to modification of the multiplier word length.

The constructive interference to be established in the bright zone is also significantly less sensitive to word length minimization, although we for this zone conclude that a reduction of the adder word length has a significantly more negative impact on the overall performance as compared to a similar reduction in the multiplier word length.

In terms of the frequency related performance, we have demonstrated that the dark zone performance is impacted essentially in the complete 50 – 600 Hz frequency band which has been subject for our investigation. Although the bright zone performance is also frequency dependent for varying word lengths, we conclude that this can be observed only to a less extent as related to PSD degradation and frequency range. Despite that a shortened adder word length reduces the bright zone performance, we found that this occurs for very limited bandwidths, primarily in the upper part of the frequency band.

Our work has shaded some previously unknown light onto the way Sound Zone Control is influenced by and potentially could be operated optimally in a reconfigurable hardware fixed-point environment. Our results are therefore of particular importance when it comes to practical realization of this emerging audio technology. Despite our many new discoveries, there are still various unsolved problems and questions needed to be answered. In our future work we therefore focus on topics related to the mapping of the control filters onto a real-time hardware platforms. In particular, we will address how essential design metrics such as execution time and power consumption potentially can be reduced when the fixed-point multipliers and adders are replaced by arithmetic units which perform their calculation approximately, [15]. Our work has shown that the sound zone application allows substantial word length reductions, as compared to a 32-bit reference, and therefore an interesting study is to investigate how arithmetic units, which introduce a certain amount of approximation for a given word length, can be operated in sound zones in order to minimize time- and power overhead in a real-time dedicated hardware architecture.

## References

[1] Stephen J. Elliott, Jordan Cheer, Harry Murfet, and Keith R. Holland. Minimally radiating sources for personal audio. *The Journal of the Acoustical Society of America*, 128, 2010. ISSN 0001-4966. doi: 10.1121/1.3479758.

[2] Jordan Cheer, Stephen J. Elliott, and Marcos F.Simón Gálvez. Design and implementation of a car cabin personal audio system. *AES: Journal of the Audio Engineering Society*, 61, 2013. ISSN 15494950.

[3] Joung-Woo Choi and Yang-Hann Kim. Generation of an acoustically bright zone with an illuminated region using multiple sources. *The Journal of the Acoustical Society of America*, 111, 2002. ISSN 0001-4966. doi: 10.1121/1.1456926.

[4] W.F Druvesteyn and J. Garas. Personal sound. *Journal of Audio Engineering*, 45, 1997.

[5] A. Canclini, D. Markovic, M. Schneider, F. Antonacci, E. A.P. Habets, A. Walther, and A. Sarti. A weighted least squares beam shaping technique for sound field control. volume 2018-April, 2018. doi: 10.1109/ICASSP.2018.8461292.

[6] Michael Buerger, Christian Hofmann, Cornelius Frankenbach, and Walter Kellermann. Multizone sound reproduction in reverberant environments using an iterative least-squares filter design method with a spatiotemporal weighting function. volume 2017-October, 2017. doi: 10.1109/WASPAA.2017.8170005.

[7] Yefeng Cai, Ming Wu, Li Liu, and Jun Yang. Time-domain acoustic contrast control design with response differential constraint in personal audio systems. *The Journal of the Acoustical Society of America*, 135, 2014. ISSN 0001-4966. doi: 10.1121/1.4874236.

[8] Yefeng Cai, Ming Wu, and Jun Yang. Design of a time-domain acoustic contrast control for broadband input signals in personal audio systems. 2013. doi: 10.1109/ICASSP.2013.6637665.

[9] Daan H.M. Schellekens, Martin B. Moller, and Martin Olsen. Time domain acoustic contrast control implementation of sound zones for low-frequency input signals. volume 2016-May, 2016. doi: 10.1109/ICASSP.2016.7471698.

[10] Martin Bo Møller and Jan Østergaard. A moving horizon framework for sound zones. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 28, 2020. ISSN 23299304. doi: 10.1109/TASLP.2019.2951995.

[11] M. Aktan, A. Yurdakul, and G. Dundar. An algorithm for the design of low-power hardware-efficient fir filters. *IEEE Trans. Circuits Syst.*, 55:1536 – 1545, 2014.

[12] Behrooz Parhami. *Computer Arithmetic, Algorithms and Hardware Design*. Oxford University Press, 2000.

[13] Martin Bo Møller and Martin Olsen. Sound zones: On envelope shaping of fir filters. *24th International Congress on Sound and Vibration, ICSV 2017*, 2017.

[14] S. Ramprasad, N. R. Shanbhag, and N. Hajj. Analytical estimation of signal transition activity from word-level statistics. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 16:718 – 733, 1997.

[15] H. Jiang, F. J. H. Santiago, H. Mo, L. Liu, and J. Han. Approximate arithmetic circuits: A survey, characterization, and recent applications. *Proceedings of The IEEE*, 108:2108 – 2135, 2020.