

# A New Fast Logarithm Algorithm Using Advanced Exponent Bit Extraction for Software-Based Ultrasound Imaging Systems

Seongjun Park <sup>1,2</sup> and Yangmo Yoo <sup>1,3,\*</sup><sup>1</sup> Department of Electronic Engineering, Sogang University, Seoul 04107, Republic of Korea<sup>2</sup> Department of Computer Science, Sogang University, Seoul 04107, Republic of Korea<sup>3</sup> Department of Biomedical Engineering, Sogang University, Seoul 04107, Republic of Korea

\* Correspondence: ymyoo@sogang.ac.kr; Tel.: +82-2-705-8907; Fax: +82-2-707-3008

**Abstract:** Ultrasound B-mode imaging provides anatomical images of the body with a high resolution and frame rate. Recently, to improve its flexibility, most ultrasound signal and image processing modules in modern ultrasound B-mode imaging systems have been implemented in software. In a software-based B-mode imaging system, an efficient processing technique for calculating a logarithm instruction is required to support its high computational burden. In this paper, we present a new method to efficiently implement a logarithm operation based on exponent bit extraction. In the proposed method, the exponent bit field is first extracted and then some algebraic operations are applied to improve its precision. To evaluate the performance of the proposed method, the peak signal-to-noise ratio (PSNR) and the execution time were measured. The proposed efficient logarithm operation method substantially reduced the execution time, i.e., eight times, compared to direct computation while providing a PSNR of over 50 dB. These results indicate that the proposed efficient logarithm computation method can be used for lowering the computational burden in software-based ultrasound B-mode ultrasound imaging systems while improving or maintaining the image quality.

**Keywords:** ultrasound B-mode imaging; float point logarithm; software-based ultrasound imaging systems; single instruction multiple data



**Citation:** Park, S.; Yoo, Y. A New Fast Logarithm Algorithm Using Advanced Exponent Bit Extraction for Software-Based Ultrasound Imaging Systems. *Electronics* **2023**, *12*, 170. <https://doi.org/10.3390/electronics12010170>

Academic Editors: Sanjay Misra, Robertas Damaševičius and Bharti Suri

Received: 9 November 2022

Revised: 24 December 2022

Accepted: 29 December 2022

Published: 30 December 2022



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

A medical ultrasound imaging system can show the anatomical structure of the body in real time. The reconstruction of ultrasound images has traditionally been implemented using hardware-based signal and image processing engines. Recently, to enhance their computational flexibility, research on software-based ultrasound imaging systems has been actively conducted with various advanced computing technologies [1–3]. For software-based ultrasound systems, an efficient software implementation method is necessary for supporting the complex signal and image processing blocks demanding high computational power.

In ultrasound B-mode imaging, log compression has been widely used for emphasizing weak scattering signals on the same scale as strong specular reflections [4]. In log compression, the envelope signal, after receive beamforming and quadrature demodulation, is transformed by applying a logarithm operation. Since the computation of logarithm operations generally requires complex floating operation on software-based ultrasound imaging systems, it takes a longer execution time. Therefore, to implement a computationally efficient software-based ultrasound imaging system, it is necessary to minimize the execution time of the logarithm operation.

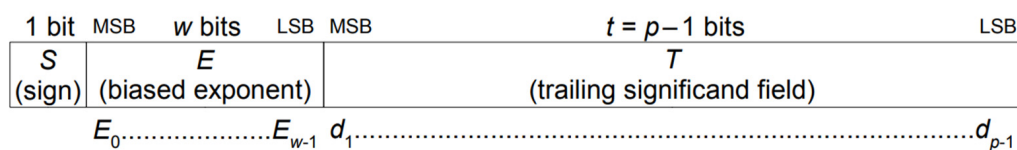
To lower the computational complexity in logarithm operations, various approximation methods, such as a look-up table (LUT) and Taylor's series, have been proposed [5,6]. In this study, to further lower this computational complexity, a new approximation method, in which the exponent bits of IEEE 754 floating-point envelope data are utilized, is presented. The IEEE 754 floating-point format is the standard notation used to represent

floating-point data in computers [7]. According to this notation, certain bits are assigned to indicate the exponent of a given floating point value. In this study, those bits for exponent are used to approximate the logarithm operation. In addition, to further lower the execution time for logarithm operations, single instruction and multiple data (SIMD) programming techniques were applied. The feasibility of the proposed logarithm approximation method was demonstrated by implementing it with an ARM Corted-A9 processor (Arm Holdings PLC, Cambridge, UK) embedded in a commercial system-on-chip (SoC) board (XC7Z020-CLG484-1, Xilinx Inc., San Jose, CA, USA).

**2. Methods**

*2.1. IEEE 754 Format*

Computers generally use the floating-point standard notation for real numbers. Currently, the most widely used floating-point standard is the IEEE 754 standard. As shown in Figure 1, this standard uses 3 bit fields for one real number expression [7].



**Figure 1.** Diagram of IEEE 754 standard floating-point’s bit field.

Each bit field represents the sign, exponent, and mantissa of the real number. Thus, a certain real number (i.e.,  $v$ ) can be expressed by the following [7]:

$$v = (-1)^S \times 2^{E-bias} \times (1 + 2^{1-p} \times T) \tag{1}$$

where  $bias$  is a constant determined by the standard, and 127 is used for the C language float type on a 64-bit AP. When the given real number  $v$  is positive—such as the envelope signal in ultrasound B-mode imaging—Equation (1), after taking the log of both sides, can be rewritten by the following:

$$\log_2(v) = (E - bias) + \log_2(1 + 2^{1-p} * T) \tag{2}$$

where  $T$  is composed of  $p-1$  bits. Since the second term on the right side of Equation (2) is less than 1, after applying a rounding down operation (i.e., *RoundDown*), Equation (2) can be represented by the following:

$$RoundDown(\log_2(v)) = E - bias \tag{3}$$

As shown in Equation (3), therefore, if a real number is expressed based on the IEEE 754 standard, the exponent bit has an approximate relation to the result of the logarithm operation. In the proposed logarithm approximation method, these relationships are utilized.

*2.2. Proposed Method: Logarithm Computation with Advanced Exponent Bit Extraction*

The log compression process in ultrasound B-mode imaging requires several considerations compared to a general logarithm operation. First, since the ultrasound B-mode imaging system performs log compression on the envelope of the received signal, the input is always positive. In addition, log-compressed signals are expressed as an eight-bit integer for display after applying a discard operation [8]. Therefore, it is possible to approximate the logarithm operation to show an appropriate level of accuracy without considering the sign. By considering these characteristics, in the proposed method, before extracting the exponent bit in Equation (3), additional processes are needed for improving its accuracy and running speed on an AP.

Figure 2 shows the block diagram of the processing modules in the proposed method. Here, input floating-point data are first parallelized for fast processing of operations. Then, the exponent bit of each data piece is separated to prevent overflow in the following processing steps. To enhance the numerical accuracy of the proposed method, an additional power operation is performed on the separated mantissa. Finally, the logarithm operation is completed by extracting the exponent bit. Each step in the proposed method will be explained in the following sub-sections.

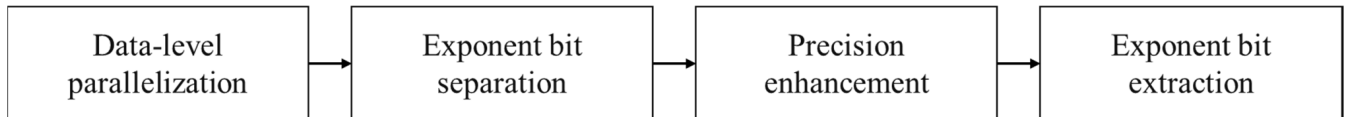


Figure 2. Block diagram of the processing modules in the proposed method.

2.3. Data-Level Parallelization

Since the proposed method performs powering operations on the input number, the iterative multiplication operation of the input data takes the main computing time. Therefore, to utilize SIMD operations supported in modern APs, multiple input data are merged to form a single vector register, and these multiple data can be processed in one instruction processing time [9]. Thus, the multiplication operation is processed through data-level parallelism, and the efficiency of the operation can be increased.

2.4. Exponent Bit Separation

The proposed method raises the parallelized input data to the power of the data to enhance the numerical resolution. Due to this powering operation, an overflow issue may occur in the 32-bit floating-point precision range. Moreover, the use of the 64-bit double precision data type may result in a performance drop due to a low data parallelization ratio. Therefore, by separating the exponent bit field, overflow in the subsequent power operation can be prevented. By taking the  $N$  square operation on both sides, Equation (1) can be rewritten as follows:

$$v^N = 2^{N \times (E - bias)} \times (1 + 2^{1-p} \times T)^N \tag{4}$$

As indicated in Equation (4), as  $N$  increases, the number of power terms of two becomes too large for the single precision representation. However, the second term on the right-hand side is less than the  $N$ th power of two. Therefore, by separating the exponent bit field of the parallelized input data, the power is performed only on the second term on the right side. The bias in Equation (4) is written to the exponent part to make the semantic value of the exponent part 0, and then, only the remaining mantissa part is  $N$ -times squared. In addition, the separated exponent part is multiplied by  $N$ . By doing so, the operation equivalent to the  $N$  square of the input data can be performed in the single precision expression.

2.5. Precision Enhancement

In ultrasound B-mode imaging systems, the result of the log operation is converted into decibel units to reconstruct an image [10,11]. Thus, the envelope signal ( $v$ ) in Equation (1) can be represented in decibel units by

$$dB(v) = 20 \log_{10}(v) \tag{5}$$

However, since the exponent bit extracted by Equation (3) is a round-down value after applying a  $\log_2()$  operation, Equation (5) can be rewritten as

$$dB(v) \approx RoundDown(\log_2(v)) \times \log_{10} 2 \times 20 = (E - bias) \times \log_{10} 2 \times 20 \tag{6}$$

Then, in ultrasound B-mode imaging systems, the dynamic range (DR) of the signal in Equation (6) is typically limited to be displayed, e.g., 40–60 dB. For a given DR value, when the maximum decibel value of the signal to be expressed by ultrasound B-mode imaging systems is assumed to be  $dB_{max}$ , Equation (6) can be rewritten as

$$\max\left(\min\left(\frac{\text{RoundDown}(\log_2(v)) \times 20\log_{10}2 - dB_{max}}{DR}, 0\right) + 1, 0\right) \quad (7)$$

From Equation (7), all signals within the DR to be expressed are mapped from 0 to 1. Regarding this value, in order to display it on a display using a n-bit integer data type, the brightness value of each pixel is calculated by remapping the result of Equation (7) to  $0 \sim (2^n - 1)$  as

$$\max\left(\min\left(\frac{\text{RoundDown}(\log_2(v)) \times 20\log_{10}2 - dB_{max}}{DR}, 0\right) + 1, 0\right) \times (2^n - 1) \quad (8)$$

The pixel brightness value calculated by Equation (8) has a discrete value. This is because the brightness value of the pixel was derived from the value of Equation (3) having a discrete value. In addition, the discrete interval is changed as the value of Equation (3), in which one is a discrete interval, is multiplied by a certain number.

The discrete spacing of pixel values is the total product of coefficients multiplied by the  $\log_2()$  truncated value in Equation (8) as follows:

$$\frac{20\log_{10}2 \times (2^n - 1)}{DR} \quad (9)$$

As seen in Equation (9), this is independent of  $dB_{max}$ . In general, since the ultrasound B-mode imaging system uses eight as value of n and 60 as the value of DR, the quantized discrete interval of the image output calculated by substituting this value into Equation (9) is 25.68. According to the change of the envelope signal, the brightness of each pixel shows only a discrete change of 25.68 in 256 steps and has quantization error. The detained change of soft tissues appears as a numerical difference smaller than this discrete interval of 25.68 on the display. Therefore, such low numerical resolution is insufficient to express detailed changes in soft tissues.

As a solution to the quantization error issue, before extracting the exponent bit, as shown in Equation (4), the envelope signal ( $v$ ) is  $N$  squared and then compensated by division after the round down operation so that the size of the discretization step in Equation (9) can be reduced. Thus, Equation (6) can be rewritten as

$$dB(v) \approx \text{RoundDown}(\log_2(v^N)) / N \times \log_{10} 2 \times 20 \quad (10)$$

Then, Equation (8) representing the brightness value of a pixel can be represented by

$$\max\left(\min\left(\frac{\text{RoundDown}(\log_2(v^N)) / N \times 20\log_{10}2 - dB_{max}}{DR}, 0\right) + 1, 0\right) \times (2^n - 1) \quad (11)$$

Finally, Equation (9) showing the size of the pixel brightness value quantization interval is given as

$$\frac{20\log_{10}2 \times (2^n - 1)}{DR \times N} \quad (12)$$

As seen in Equation (12),  $N$  reduces the output quantization error. However, since the input must be multiplied by the size of  $N$ , the complexity of the operation and the size of  $N$  are inversely proportional. Therefore, it is necessary to select an appropriate  $N$  value.

In this study, 32 was selected as the value of  $N$  as a compromised value of these trade-off relationship. This is in consideration of the characteristics of a display monitor, in which brightness values for each pixel are stored and expressed as an integer data type.

In addition, when  $N$  is larger than 32, the size of the pixel brightness value quantization interval in Equation (12) is smaller than 1 so that all integer data can be used on the display without a gap. However, since the exponent bit separation technique in Section 2.4 is used, only the separated mantissa (fraction bit) is raised to the power of 32. The exponent bit must be multiplied by 32, but, as shown in Equation (9), it is divided by 32 again so that it is not necessary to multiply by 32.

### 2.6. Exponent Bit Extraction

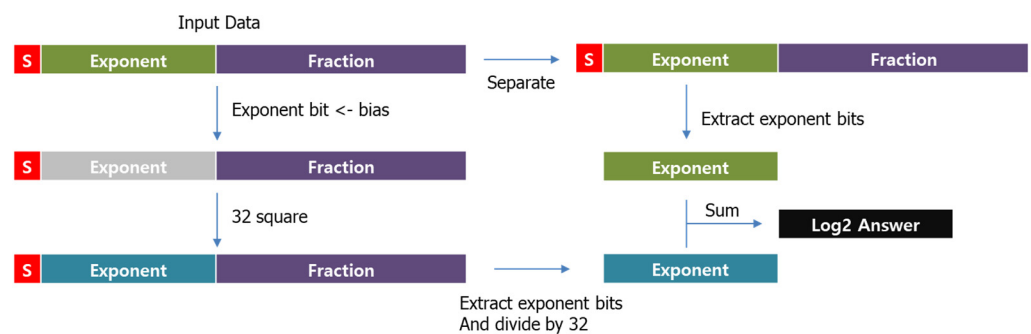
Now, the input real number is divided into an exponent part separated by exponent bit separation and a mantissa part to the power of 32. Meanwhile, if a real number is multiplied by 32 and then  $\log_2$  is taken, it is expressed by the following:

$$\log_2 v^{32} = 32(E - bias) + \log_2(1 + 2^{1-p} * T)^{32} \tag{13}$$

Then, after performing rounding and dividing by 32, Equation (5) can be represented by the following:

$$fix(\log_2 v^{32})/32 = (E - bias) + fix(\log_2(1 + 2^{1-p} * T)^{32})/32 \tag{14}$$

The first term on the right side of Equation (6) is the exponential part separated by exponent bit separation. Then, the second term is equivalent to the mantissa of the input after being powered, exponent-bit-extracted, and divided by 32. Therefore, by adding two values, it is possible to approximate the logarithm operation more accurately and efficiently. All these operations are performed by the SIMD pipeline as described in Figure 3.



**Figure 3.** SIMD operation for logarithm operation by exponential division and power in the proposed method.

### 2.7. Experimental Setup

To evaluate the efficiency and accuracy of the proposed method, phantom experiments were conducted. In the experiment, two phantom data pieces were obtained by the Vantage™ Research Ultrasound Systems (Verasonics Inc., Kirkland, WA, USA) from two tissue mimicking phantoms (CIRS, SUN NUCLEAR, Louisa, VA, USA). These data were processed with the system-on-chip (SoC) evaluation board (XC7Z020-CLG484-1, Xilinx Inc.) equipped with an ARM Cortex-A9 AP (Arm Holdings PLC, Cambridge, UK). The SIMD operation was implemented with the NEON intrinsic provided by the ARM processor. The code implementing the proposed method was compiled by the GNU compiler collection (GCC).

The accuracy was compared with the log function provided by the math library of the GCC. In addition, the peak signal-to-noise-ratio (PSNR) was computed as follows [12]:

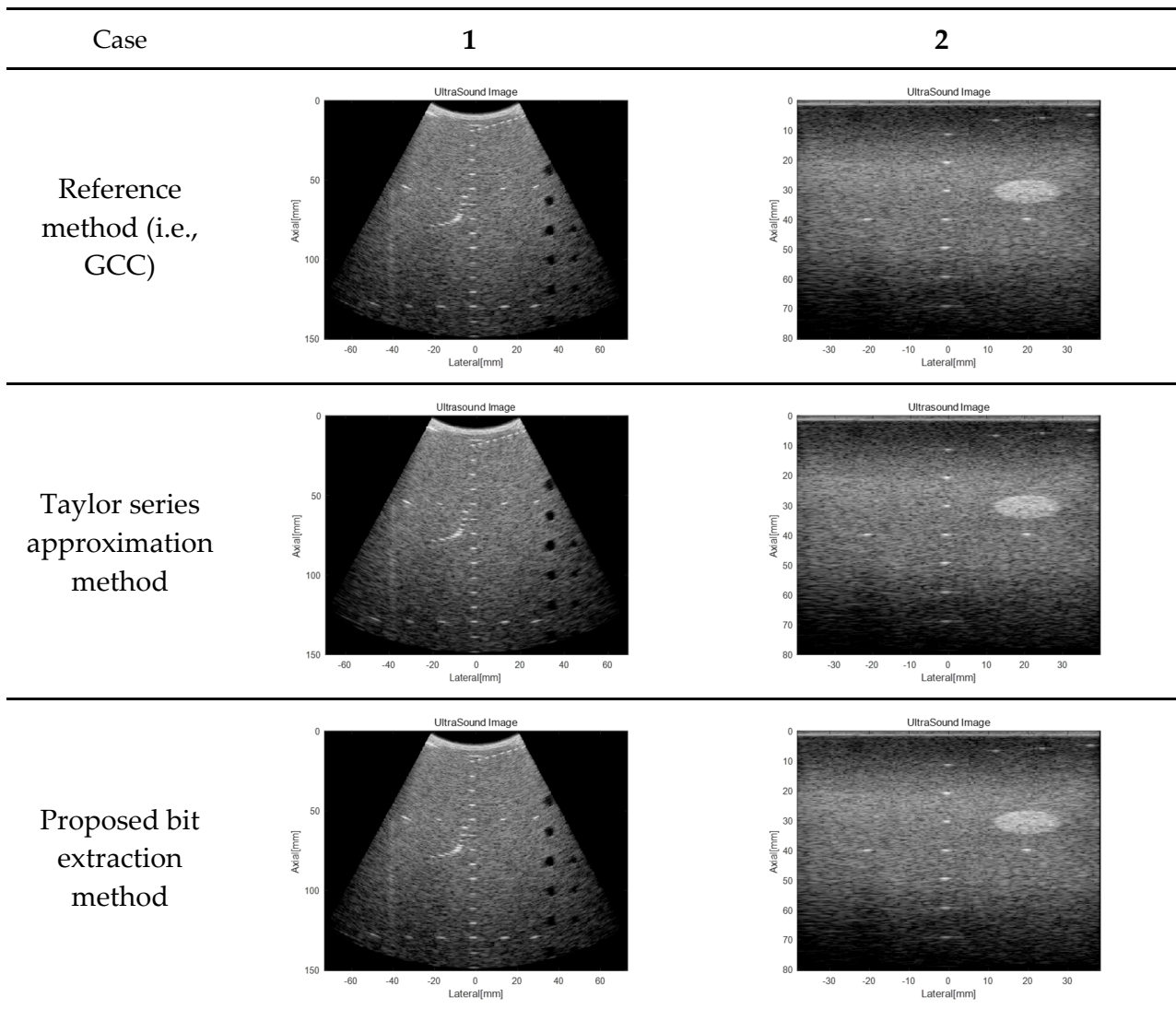
$$PSNR = 10 \log_{10} \left( \frac{MAX_i^2}{MSE} \right) \tag{15}$$

where  $MAX_i$  is the maximum value of the image and  $MSE$  is the mean squared error between two images from the reference and proposed methods. The processing time was

also measured by using the time measurement function of the ARM processor for input data with 192 scanlines and 512 samples.

### 3. Experiment Results

Figure 4 shows the ultrasound B-mode images from two tissue mimicking phantoms when using the reference (i.e., math library in GCC) and the proposed bit extraction methods along with the Taylor series approximation method. As shown in Figure 4, under visual assessment, it is difficult to identify the difference among the three logarithm computation methods.



**Figure 4.** Comparison of reconstructed images of two phantom data pieces from the reference, Taylor series approximation and proposed bit extraction methods.

For quantitative comparison, the PSNR values for the Taylor series approximation and proposed bit extraction methods were computed, as described in Equation (5). As summarized in Table 1, the proposed bit extraction method presented 55.2 dB and 58.2 dB for ultrasound B-mode images when using a convex array probe and a linear array probe, respectively. Although the PNSR values from the proposed bit extraction method are lower than those from the Taylor series approximation method (i.e., 96.3 dB and 99.6 dB), as demonstrated in Figure 4, the proposed bit extraction method is suitable for being used in ultrasound B-mode imaging.

**Table 1.** Comparison of PSNR values of Taylor series approximation and proposed bit extraction methods with two phantom data sets.

Case	1	2
Taylor series approximation method	96.3 dB	99.6 dB
Proposed bit extraction method	55.2 dB	58.2 dB

The execution times for the approximation methods, including the Taylor series approximation and proposed bit extraction methods, were compared with the reference method as summarized in Table 2. The proposed bit extraction method was eight-fold faster than the reference where the built-in function (i.e.,  $\log_{10}$ ) in GCC was utilized. Moreover, it outperformed the Taylor series approximation method, at 2.8 ms vs. 10.1 ms, respectively. Both approximation methods (i.e., Taylor series and bit extraction) were implemented by utilizing SIMD operations in the ARM AP.

**Table 2.** Comparison of computation times from the reference, Taylor's series approximation, and proposed bit extraction methods.

	Reference (GCC)	Taylor Series (SIMD)	Bit Extraction (SIMD)
Time (ms)	23.9	10.1	2.8

#### 4. Conclusions

In this study, the log compression of ultrasound B-mode imaging was able to be optimized by using advanced exponent bit extraction. When validating its feasibility using an ARM Cortex-A9 AP, the performance of the proposed bit extraction method outperformed the reference and Taylor's series methods in terms of execution time. Moreover, the proposed bit extraction method showed a PSNR value over 55 dB, which is reasonable in ultrasound B-mode imaging. Thus, the proposed efficient logarithm computation method based on bit extraction can enhance the performance of software-based ultrasound B-mode imaging systems while preserving image quality.

**Author Contributions:** Conceptualization, S.P. and Y.Y.; methodology, S.P.; validation, S.P. and Y.Y.; writing—original draft preparation, S.P.; writing—review and editing, S.P. and Y.Y.; supervision, Y.Y.; funding acquisition, Y.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2021R1A2C3006264), the Korea Medical Device Development Fund grant funded by the Korean government (the Ministry of Science and ICT, the Ministry of Trade, Industry and Energy, the Ministry of Health and Welfare, Republic of Korea, the Ministry of Food and Drug Safety) (Project Number: 202011A01).

**Data Availability Statement:** The data that support the findings of this study are available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The authors declare no conflict of interest.

#### References

- Kim, G.-D.; Yoon, C.; Kye, S.-B.; Lee, Y.; Kang, J.; Yoo, Y.; Song, T.-K. A single FPGA-based portable ultrasound imaging system for point-of-care applications. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **2012**, *59*, 1386–1394. [[CrossRef](#)] [[PubMed](#)]
- Kerem, K.; Lee, C.; Kim, Y. Software-based ultrasound beamforming on multi-core DSPs. In Proceedings of the IEEE International Ultrasonics Symposium, Orlando, FL, USA, 18–21 October 2011; pp. 503–506.
- Boni, E.; Yu, A.C.H.; Freear, S.; Jensen, J.A.; Tortoli, P. Ultrasound Open Platforms for Next-Generation Imaging Technique Development. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **2018**, *65*, 1078–1092. [[CrossRef](#)] [[PubMed](#)]
- Kossoff, G. Basic Physics and Imaging Characteristics of Ultrasound. *World J. Surg.* **2000**, *24*, 134–142. [[CrossRef](#)] [[PubMed](#)]
- Brubaker, T.; Becker, J. Multiplication Using Logarithms Implemented with Read-Only Memory. *IEEE Trans. Comput.* **1975**, *100*, 761–765. [[CrossRef](#)]

6. Ucker, P.; Weirich, M.R.; Paim, G.; da Costa, E.A.C.; Bampi, S. Optimizing Iterative-based Dividers for an Efficient Natural Logarithm Operator Design. In Proceedings of the 2020 IEEE 11th Latin American Symposium on Circuits & Systems (LASCAS), San Jose, Costa Rica, 25–28 February 2020; pp. 1–4. [[CrossRef](#)]
7. *ANSI/IEEE Standard 754-1985*; IEEE Standard for Binary Floating-Point Arithmetic. Institute of Electrical and Electronic Engineers: New York, NY, USA, 1985.
8. York, G.; Kim, Y. Ultrasound Processing and Computing: Review and Future Directions. *Annu. Rev. Biomed. Eng.* **1999**, *1*, 559–588. [[CrossRef](#)] [[PubMed](#)]
9. Hassaballah, M.; Omran, S.; Mahdy, Y.B. A Review of SIMD Multimedia Extensions and their Usage in Scientific and Engineering Applications. *Comput. J.* **2008**, *51*, 630–649. [[CrossRef](#)]
10. Busey, H.W.; Rosenblum, L.H. Physical Aspects of Gray Scale Ultrasound. In *Ultrasound in Medicine*; White, D., Ed.; Springer: Boston, MA, USA, 1975; pp. 559–566. [[CrossRef](#)]
11. Diouf, M.; Lin, Z.; Harling, M.; Toussaint, K.C. Demonstration of speckle resistance using space–time light sheets. *Sci. Rep.* **2022**, *12*, 14064. [[CrossRef](#)] [[PubMed](#)]
12. Horé, A.; Ziou, D. Image Quality Metrics: PSNR vs. SSIM. In Proceedings of the 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.