# VLSI architectures for Delay Multiply and Sum Beamforming in Ultrasound Medical Imaging

Gayathri Malamal and Mahesh Raveendranatha Panicker
*Center for Computational Imaging*
*Indian Institute of Technology Palakkad*,
Kerala, India
{121814001@smail.iitpkd.ac.in, mahesh@iitpkd.ac.in}

***Abstract*—Ultrasound medical imaging systems typically follow standard delay and sum (DAS) beamforming at the reception for image reconstruction. In DAS, the echo signals that are returned to the transducer are aligned in time and summed to form the beamformed signal. To improve the image quality and the signal to noise ratio of DAS, a non-linear beamforming named delay multiply and sum (DMAS) has been proposed in the literature, where, the signals arriving at the transducer are aligned in time and are pairwise multiplied in all possible combinations before summation. This provides better coherence, a correlation-based data-driven apodization, and consequently result in better contrast and resolution. However, the computational complexity of DMAS is higher than DAS thus restricting its real-time implementation. This paper presents two novel VLSI architectures for the implementation of DMAS, whose complexity is independent of the number of transducer elements. The proposed architectures are implemented on xc7z010clg400-1 FPGA and the results clearly show the channel independency of the proposed architectures.**

***Keywords—Ultrasound imaging, beamforming, delay multiply and sum, FPGA***

## I. INTRODUCTION

Ultrasound (US) is one of the most preferred medical imaging modalities due to the lack of ionizing radiation, easy portability, low cost, and repeatability. Typically, US imaging systems follow a pulse echo approach of imaging, where the US signals are sent into the tissue from an array of transducer elements with a fixed center frequency and collect back the reflected/scattered signals from the tissue using the same set of transducer elements [1]. The reflected signals (radio frequency (RF) signals) that are detected by the transducer array elements are gain compensated as a function of time as the signals from larger depths will be subjected to higher attenuation. The signals which are then digitized by an analog to digital converter (ADC) are beamformed by a suitable beamforming scheme. Further, the beamformed signals are demodulated (in-phase quadrature), envelope extracted, and log compressed for reconstructing a diagnosable image [2].

The algorithm that is adopted to beamform the digitized RF data plays an integral part in deciding the quality of the final image. Among several beamforming algorithms proposed in the literature, the most commonly employed is the delay and sum (DAS) due to its simplicity in real-time implementation. In DAS, each of the points at the receive is dynamically reconstructed by summing the delay compensated digitized RF signals. The delay compensation is necessary to accurately align each of the digitized RF signals from the transducer array elements/channels in time [3]. However, the DAS reconstruction follows a data independent addition based on geometric delays and does not result in a superior image. The images suffer from a reduced contrast and resolution due to very low signal to noise ratio and increased side lobe levels. To overcome these challenges, several data dependent beamforming algorithms have also been proposed, which continuously update the aperture weights according to the receive data and thereby suppressing the off-axis signals. But these algorithms are computationally intensive and do not favor a straightforward real-time implementation [4]-[7]. Another beamforming algorithm, namely delay multiply and sum (DMAS) has been proposed to provide better contrast and resolution than DAS which has been originally devised for breast cancer detection using microwave imaging [8] and later tailored to fit into US imaging as in [9]. The concept of DMAS does not significantly vary from DAS. In DMAS, the delay compensated digitized RF signals from the channels are pairwise multiplied in all possible combinations and then summed together to form the final beamformed signal.

However, due to the computational complexity introduced majorly by the pairwise multiplications, DMAS still has not been able to displace DAS from commercial systems. But with the advancement in electronics which includes the introduction of high-speed field-programmable gate arrays (FPGAs), there is a possibility towards this. Although software-based beamforming using CPUs or GPUs is popular, they require an enormous amount of raw RF data to be transferred for processing beamforming and may not be suitable for real-time implementation. With the FPGAs providing dedicated reconfigurable and reusable logic resources, there is viability for an optimized real-time implementation of DMAS [10]. To the best of our knowledge, very little work could be seen towards developing comprehensive hardware-oriented architectures specifically emphasizing digital beamforming. The existing works are majorly confined to a superficial aspect of the front to backend implementation rather than an exclusive block specific implementation. However, beamforming specific hardware architectures could result in better logic optimization and also aid in developing reconfigurable pixel-level beamforming approaches as proposed by us in [11].

In this work, two novel VLSI architectures based on the mathematical formulations introduced for DMAS in [12] and [13] are proposed for possible real-time implementation. The hardware footprint is analyzed by implementing the architectures on Xilinx xc7z010clg400-1 FPGA and timing complexity is estimated through Xilinx Vivado simulations.

The paper is organized as follows. Section II describes the DMAS algorithm and its mathematical variants that have been proposed for enabling real-time implementation. The proposed VLSI architectures are presented in Section III. Section IV explains the experimental setup. Section V discusses FPGA synthesis and timing results and the conclusions are presented in Section VI.

## II. BACKGROUND

This section describes the existing architectures of DMAS in detail. DMAS is a non-linear data independent

978-1-7281-8895-9/20/$31.00 ©2020 IEEE

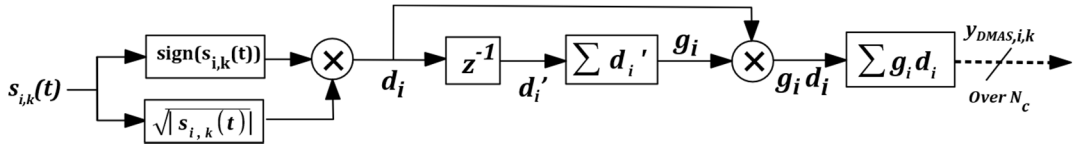


Fig. 1. FBA schematic illustration (The dashed lines indicate that cumulative is getting transferred from the source to destination)

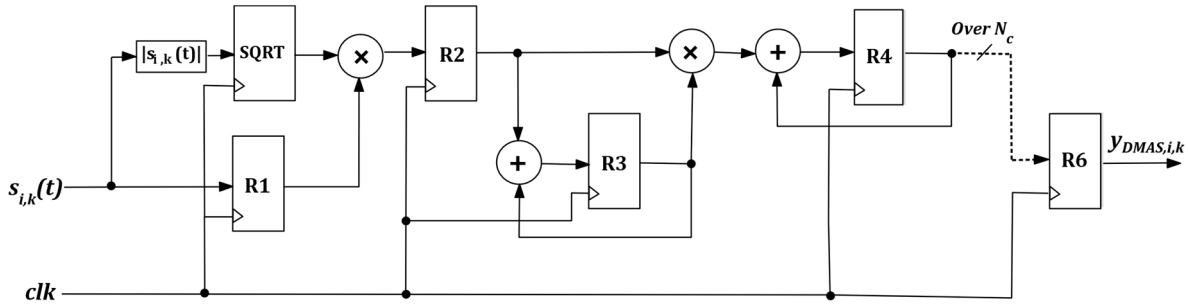


Fig. 2. FBA RTL architecture (The dashed lines indicate that cumulative is getting transferred from the source to destination)

beamforming algorithm used for US image reconstruction. DMAS introduces an extra multiplication stage to DAS where the delay compensated RF signals are multiplied in all possible combinations (except the self-products) before summing to form the final beamformed signal as in [9]. This depicts a kind of correlation leading to better coherence by providing a measure of how a point on the receive is viewed by different transducer array elements/channels. This results in better noise rejection thereby improving the contrast. The DMAS beamformed output $y_{DMAS,k}(t)$, could be expressed as below,

$$y_{DMAS,k}(t) = \sum_{i=1}^{N_c-1} \sum_{j=i+1}^{N_c} sign\left(s_i(t)s_j(t)\right) \sqrt{\left|s_i(t)s_j(t)\right|} \quad (1)$$

where $k$ is the beamformed pixel or scanline in the scan area $S$, $N_c$ is the total number of transducer array elements/channels, $s_{i,k}$ , $s_{j,k}$ are the delay compensated signals of the $i^{th}$ and $(i+1)^{th}$ array element or channel respectively. The square root operation is required here as otherwise would result in a dimensionally squared signal of non-zero mean which cannot be subjected to envelope detection. The delay compensated signals having a similar frequency spectrum when multiplied would produce baseband and harmonic components in the spectrum of the beamformed signal. In [9], the DMAS beamformed signal is further filtered to isolate the second harmonic and reject the baseband to provide a better resolution than DAS. However, the RF data should be sufficiently oversampled to generate the harmonic frequencies.

However, the prime reason for the increase in the computational complexity of DMAS is the presence of combinatorial multiplications. The implementation in (1) requires $N_c(N_c-1)/2$ multiplications, elevating the computational complexity to $O({N_c}^2)$ as opposed to $O(N_c)$ of DAS which is a serious concern for a real-time implementation as the computational complexity would increase quadratically with the number of array elements/channels. To address this, two new approaches in [12] and [13] have been proposed. The approach in [12] reduces the computational complexity by factorizing the expression in (1) as,

$$\begin{aligned} y_{DMAS,k}(t) = {} & s_{1,k}'(t)[s_{2,k}'(t) + s_{3,k}'(t) + \cdots + s_{N_c,k}'(t)] + \\ & s_{2,k}'(t)[s_{3,k}'(t) + s_{4,k}'(t) + \cdots + s_{N_c,k}'(t)] + \\ & \ldots + s_{N_c-2,k}'(t)[s_{N_c-1,k}'(t) + s_{N_c,k}'(t)] + \\ & s_{N_c-1,k}'(t)s_{N_c,k}'(t) \end{aligned} \quad (2)$$

where,

$$s_{i,k}'(t) = sign(s_{i,k}(t))\sqrt{|s_{i,k}(t)|} \quad (3)$$

$$s_{j,k}'(t) = sign(s_{j,k}(t))\sqrt{|s_{j,k}(t)|} \quad (4)$$

This approach brings down the number of multiplications to $N_c - 1$. Similarly, in [13] a mathematically simplified approach of DMAS is proposed. This method uses a reformulation of DMAS in (1) using multinomial theorem for the power of 2 to obtain,

$$y_{DMAS,k}(t) = \frac{1}{2}[A - B] \quad (5)$$

where,

$$A = \left(\sum_{i=1}^{N_c} s_{i,k}'(t)\right)^2 \quad (6)$$

$$B = \sum_{i=1}^{N_c} |s_{i,k}(t)| \quad (7)$$

The expression in (5) reduces the number of multiplications to 1 as it just replicates the pairwise multiplications by an even simpler equivalent mathematical expression.

## III. Proposed Architectures

In this section, we propose two novel VLSI architectures for the implementation of DMAS based on (2) and (5) for pixel-level beamforming. The first architecture which is based on (2) is called a factor based architecture (FBA) and the second which is based on (5) is called a multinomial theorem based architecture (MTBA). The architectures follow sequential processing of RF data where the data from the channels are accessed sequentially in the order of arrival rather

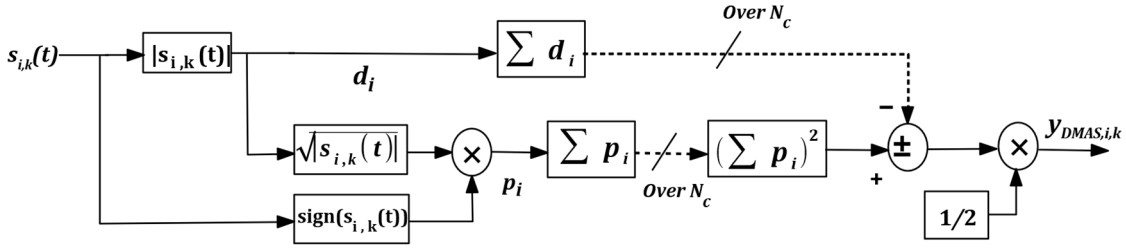


Fig. 3. MTBA schematic illustration (The dashed lines indicate that cumulative is getting transferred from the source to destination)

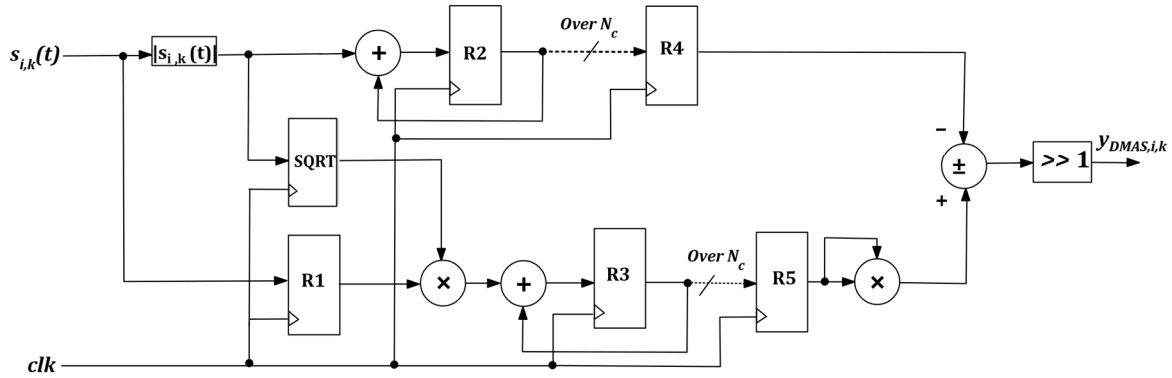


Fig. 4. MTBA RTL architecture (The dashed lines indicate that cumulative is getting transferred from the source to destination)

than a parallel approach of processing channels which consequently make the architecture not heavily dependent on the number of array elements/channels. The proposed architectures support I/Q data as well but in this work, illustrations are shown with RF data.

### *A. Factor based architecture (FBA)*

The sequential approach for reconstructing a single pixel using FBA is schematically illustrated in Fig. 1. Initially, the sign bit is extracted from the pre-beamformed oversampled RF data ($s_{i,k}(t)$) and the absolute value $|s_{i,k}(t)|$ is obtained separately to compute the square root. Further, $\sqrt{|s_{i,k}(t)|}$ is multiplied with the already extracted sign bit to form $d_i$. To obtain the factors as in (2), $d_i'$ (which are delayed $d_i$s) are summed in the order of arriving RF data to form $g_i$s and are multiplied with the next incoming channel data $d_i$ simultaneously. Once the $(N_c - 1)$ factors are generated, they are then summed to obtain $y_{DMAS,k}(t)$.

The VLSI register transfer level (RTL) representation for FBA is presented in Fig. 2. The register R1 stores the sign bit extracted from $s_{i,k}(t)$ and $\sqrt{|s_{i,k}(t)|}$ is stored in the SQRT register. The product $sign\left(s_{i,k}(t)\right)\sqrt{|s_{i,k}(t)|}$ in register R2 and is added to register R3 recursively in every clock cycle with each incoming RF data. Simultaneously, R3 is multiplied with R2 to generate the factors in (2) and the product is iteratively added to register R4 every clock cycle until the sum of all the $(N_c - 1)$ factors are cumulated in R4 to obtain the final beamformed output $y_{DMAS,k}$.

### *B. Multinomial theorem based architecture (MTBA)*

The sequential approach used in MTBA is schematically illustrated in Fig. 3. The initial computations which are the sign bit extraction and the computation of $\sqrt{|s_{i,k}(t)|}$ are similar as in FBA. The $p_i$ which is the signed square root, $sign\left(s_{i,k}(t)\right)\sqrt{|s_{i,k}(t)|}$ is summed over $N_c$ array elements/channels and is squared to generate $A$ in (5). Simultaneously, the $d_i$s which are $|s_{i,k}(t)|$ summed over the $N_c$ channels to compute $B$. Further, the difference between $A$ and $B$ is calculated followed by a division by 2 to obtain $y_{DMAS,k}$.

The VLSI RTL representation of MTBA is presented in Fig. 4. As in FBA, the register R1 stores the sign bit extracted from $s_{i,k}(t)$ and SQRT stores $\sqrt{|s_{i,k}(t)|}$. The product, $sign\left(s_{i,k}(t)\right)\sqrt{|s_{i,k}(t)|}$ is recursively added to R3 and the cumulative sum over $N_c$ array elements/channels are obtained in R5. Simultaneously, $|s_{i,k}|$ is accumulated in R2 every clock cycle and the cumulative sum over $N_c$ array elements/channels are stored in R4. The content of R5 is squared to obtain $A$ from which the content of R4 which is $B$ is subtracted. The final division by 2 is achieved with an equivalent single bit right shift operation rather than a complex division or multiplication to yield $y_{DMAS,k}$ as the final beamformed signal of the corresponding pixel.

## IV. Experimental Setup

The proposed architectures were implemented in Xilinx xc7z010clg400-1 FPGA (Xilinx Inc. San Jose, CA) [14] using the Zybo development board [15]. The logic utilization and timing results were estimated with Xilinx Vivado 2019.1. The internal clock of frequency 125 MHz was used for the implementation. The beamforming accuracy was verified by synthesizing the architectures using RF data available from Plane-Wave Imaging Challenge in Medical Ultrasound (PICMUS) [16], [17]. The synthesis and timing results are presented for $N_c = 64, 128,$ and $256$.

TABLE I
XC7Z010CLG400-1 FPGA IMPLEMENTATION AND TIMING COMPARISON FOR DIFFERENT $N_c$

| Architecture | $N_c$ | Logic LUTs (%) | Memory LUTs (%) | Registers (%) | Block RAMs (%) | DSPs (%) | No. of clock cycles |
|---|---|---|---|---|---|---|---|
| **FBA** | 64 | 1.24 | 0.13 | 0.48 | 0 | 1.25 | 70 |
| | 128 | 1.26 | 0.13 | 0.49 | 0 | 1.25 | 134 |
| | 256 | 1.25 | 0.13 | 0.50 | 0 | 1.25 | 262 |
| **MTBA** | 64 | 1.33 | 0.13 | 0.57 | 0 | 1.25 | 70 |
| | 128 | 1.34 | 0.13 | 0.57 | 0 | 1.25 | 134 |
| | 256 | 1.36 | 0.13 | 0.58 | 0 | 1.25 | 262 |

The RF data samples were aligned in time and apodized in MATLAB and the corresponding 16-bit signed fixed-point representation was used for RTL implementation. The architectures were coded in Verilog. The 16-bit representation was adopted assuming the general case where ADC samples the analog echoes with a 16-bit resolution and also this would avoid the precision loss during truncation in square root computation. The square root computations in the architectures were computed using Xilinx CORDIC IP core with optimal pipelining with a latency of $t_{SQRT}$ . The CORDIC method of square root computation was chosen because it does not introduce any extra multiplication and computes the square root with shift and add operations [18]. To compensate for the latency of the square root operation, R1 was essentially implemented as a shift register. In both the architectures, the accumulated transfers over the $N_c$ array elements/channels were controlled using counters.

The timing complexity of the architectures was estimated from the number of clock cycles required to complete the beamforming process of a single pixel. As the proposed architectures follow sequential processing of channels as discussed in Section III, the timing complexity would be a function of the number of array elements/channels. Therefore, the total number of clock cycles could be expressed as,

$$t_{DMAS} = t_{SQRT} + N_c + 1 \quad (8)$$

where, $t_{DMAS}$ is the total number of clock cycles required for the beamforming process, $t_{SQRT}$ is the square root computation latency. It has to be noted that, this can be attributed to the pipelined fashion of the data flow in the architectures.

## V. RESULTS AND DISCUSSIONS

This section discusses the results obtained from synthesizing the architectures on FBA and MTBA.

### *1) Logic Utilization*

The implementation results for three different numbers of array elements/channels are summarized in Table I. The percentages are determined with the help of total logic count available in [13] of Xilinx Zynq device xc7z010clg400-1. It is observed that the increase in channel number does not significantly scale up the resource requirement for the architectures. However, the negligible increase is due to the extra computation required in the counters tracking the channel count for accumulated transfers. It can be seen that there is a small utilization of memory LUTs, which is due to the storage of precomputed interpolated RF data. The implementation results also indicate that FBA has reduced logic requirement than MTBA which is self-evident from the RTL architectures and is attributed to the presence extra addition and shifting operation in MTBA that is not required in FBA. To put more theoretical detail on this, an analysis of the processing complexity is shown in Table II in terms of the critical number of mathematical computations from (2) and (5-7) required for a single pixel reconstruction. To visualize a larger picture, a pixel grid size of 240x320 was considered for $N_c$ =128. A direct theoretical extrapolation would be an overfit on the chosen FPGA but would easily fit onto high-end FPGAs. However, the logic count is assured to go down by considering the pixel to pixel independence on delay compensation which will be dealt with as future work.

### *2) Timing Complexity*

The timing complexity is estimated from simulations by the number of clock cycles required for the beamforming process. The results for the same are presented in Table I. It could be observed that the time complexity of the architectures follows (8) indicating that the timing complexity is proportional to the number of array elements/channels. In this case, the CORDIC IP provided by Xilinx for square root computation with optimal pipelining had a latency $t_{SQRT} = 5$.

TABLE II
THEORETICAL COMPARISON OF PROCESSING COMPLEXITY IN TERMS OF MATHEMATICAL COMPUTATIONS

| Architecture | Additions | Multiplications | Square root |
|---|---|---|---|
| **FBA** | $N_c - 2$ | $N_c - 1$ | $N_c$ |
| **MTBA** | $2(N_c - 1) + 1$ | 1 | $N_c$ |

## VI. CONCLUSION

In this work, two novel RTL architectures are demonstrated for the realization of DMAS beamforming through sequential processing of array elements/channels in US medical imaging. The architectures were implemented on xc7z010clg400-1 FPGA and the hardware resource utilization is estimated using Xilinx Vivado 2019.1. The results show that the proposed architectures are independent of the number of array elements/channels though the timing complexity increases with the number of array elements/channels. The architectures presented in this work are for US imaging, the same could be applied to other applications such as RF or microwave implementing DMAS beamforming.

## REFERENCES


[1] T. Szabo, Diagnostic Ultrasound Imaging, 2nd ed. Academic Press, 2014.

[2] F. K. Schneider, A. Agarwal, Y. M. Yoo, T. Fukuoka, and Y. Kim, "A Fully Programmable Computing Architecture for Medical Ultrasound Machines," in *IEEE Trans. Inf. Technol. B.* vol. 14, no. 2, pp. 538-540, Mar. 2010.

[3] K. E. Thomenius, "Evolution of ultrasound beamformers," in *Proc. IEEE Ultrason. Symp.*, pp. 1615-1622, 1996.

[4] J. A. Mann and W. F. Walker, "A constrained adaptive beamformer for medical ultrasound: initial results," *in Proc. IEEE Int. Ultrason. Symp.*, pp. 1807-1810, Oct. 2002.

[5] M. Sasso and C. Cohen-Bacrie, "Medical ultrasound imaging using the fully adaptive beamformer," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 489-492, 2005.

[6] J. F. Synnevag, A. Austeng, and S. Holm, "Adaptive beamforming applied to medical ultrasound imaging," in *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 54, no. 8, pp. 1606-1613, Aug. 2007.

[7] B. M. Asl and A. Mahloojifar, "Minimum variance beamforming combined with adaptive coherence weighting applied to medical ultrasound imaging," in *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 56, no. 9, pp. 1923-1931, Sep. 2009.

[8] H. B. Lim, N. T. Nhung, E. P. Li, and N. D. Thang, "Confocal microwave imaging for breast cancer detection: delay-multiply-and-sum image reconstruction algorithm," in *IEEE Trans. Biomed. Eng.*, vol. 55, no. 6, pp. 1697-1704, Jun. 2008.

[9] G. Matrone, A. S. Savoia, G. Caliano, and G. Magenes, "The delay multiply and sum beamforming algorithm in ultrasound B-mode medical imaging," in *IEEE Trans. Med. Imag.*, vol. 34, no. 4, pp. 940-949, Apr. 2015.

[10] E. Boni, A. C. H. Yu, S. Freear, J. A. Jensen, and P. Tortoli, "Ultrasound Open Platforms for Next-Generation Imaging Technique Development," in *IEEE Trans. Ultrason., Ferroelectr., Freq. Control*, vol. 65, no. 7, pp. 1078-1092, Jul. 2018.

[11] G. Malamal and M. R. Panicker, "Towards A Pixel-Level Reconfigurable Digital Beamforming Core for Ultrasound Imaging," in *IEEE Trans. Biomed. Circuits Syst.* , vol. 14, no. 3, pp. 570-582, Jun. 2020.

[12] M. Mozaffarzadeh, M. Sadeghi, A. Mahloojifar, and M. Orooji, "Double-stage delay multiply and sum beamforming algorithm applied to ultrasound medical imaging", in *Ultrasound Med. Biol.*, vol. 44, no. 3, pp. 677-686, 2018.

[13] A. Ramalli *et al*., "High dynamic range ultrasound imaging with real-time Filtered-Delay Multiply and sum beamforming," in *Proc. IEEE Int. Ultrason. Symp.*, pp. 1-1, 2017.

[14] Xilinx Zynq 7000 SoC Datasheet, Jul. 2018 [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf.

[15] Digilent, "Zybo Reference Manual", 2017 [online]. Available: https://reference.digilentinc.com/reference/programmable-logic/zybo/reference-manual.

[16] A. Rodriguez-Molares, O. M. H. Rindal, O. Bernard, A. Nair, M. A. Lediju Bell, H. Liebgott, A. Austeng, and L. Løvstakken, "The ultra-sound toolbox," in *Proc. IEEE Int. Ultrason. Symp.*, pp. 1-4, Sept. 2017.

[17] Plane-wave Imaging Challenge in Medical UltraSound (PICMUS), [Online], 2016. Available: https://www.creatis.insalyon.fr/Challenge/IEEE_IUS_2016/home.

[18] J. S. Walther, "A unified algorithm for elementary functions," in *Proc. 38th Spring Joint Comput. Conf.*, pp. 379-385, May 1971.