Customized High Performance Low Power Processor for Binaural Speaker Localization

N.Behmann*, C. Seifert*, G. Paya-Vaya*, H. Blume*, P. Jääskeläinen†, J. Multanen†, H. Kultala†, J. Takala†, J. Thiemann‡ and S. van de Par‡
*Institute of Microelectronic Systems and Cluster of Excellence Hearing4All
Leibniz Universität Hannover, Hanover, Germany
{behmann, seifert, guipav,blume}@ims.uni-hannover.de
†Department of Pervasive Computing
Tampere University of Technology, Tampere, Finland
{pekka.jaaskelainen, joonas.multanen, heikki.kultala, jarmo.takala}@tut.fi
‡Dept. of Medical Physics and Acoustics and Cluster of Excellence Hearing4All
University of Oldenburg, Oldenburg, Germany
{joachim.thiemann, steven.van.de.par}@uni-oldenburg.de

***Abstract*—One of the key problems for hearing impaired persons represents the *cocktail party* scenario, in which a bilateral conversation is surrounded by other speakers and noise sources. State-of-the-art beamforming techniques are able to segregate specific sound sources from the environment, presupposing the position of the speaker. The speaker position can be estimated in the frontal azimuth-plane with a probabilistic localization algorithm from the binaural microphone input of the both-eared hearing aid system. However, the binaural speaker localization requires computationally complex audio processing and filtering. The high computational complexity combined with low energy requirements to meet the battery constraints of hearing aid devices presents an implementation challenge.**

This paper proposes a customized C programmable processor design to implement the speaker localization algorithm that fulfills the challenging requirements placed by the usage context. When compared to a VLIW-based processor design with similar basic computational resources and no special instructions, the proposed processor reaches a $151\times$ speed-up. For a $28nm$ standard CMOS technology, power consumption of $12\ mW$ (at $50\ MHz$) and silicon area of $0.3\ mm^2$ is estimated. This is the first publication of a realistic programmable processing architecture for the probabilistic binaural speaker localization or a comparably complex algorithm for hearing aid devices. The algorithms supported by the previously proposed implementations are approximately $15\times$ less computationally demanding.

I. Introduction

Beside an even greater number of people with moderate hearing loss, about 5 % of the world's population (360 million people) suffer from a disabling hearing impairment, characterized by a hearing loss greater than 40 dB [1]. One of the most challenging acoustic situations for hearing impaired persons is the so-called *cocktail party* scenario, in which two speakers focus on their conversation, while others are speaking at the same time. The initial processing step in the computational auditory scene analysis for audio source grouping and segregation represents the localization of speakers. The succeeding audio algorithms can then apply beamforming techniques in the precomputed direction or classify different speakers. A probabilistic model for robust sound source localization based on binaural input data is presented by May [2]. For a given accuracy, the algorithm requires complex mathematic calculations on audio cues, in combination with bandwidth-demanding memory accesses for the implemented gaussian mixture model classificator.

Choosing the processing architecture for future hearing aid devices poses challenges as the developer has to fulfill multiple strict implementation requirements simultaneously. In order to keep the battery life time in an usable level, low power consumption is the highest prioritized design requirement while low latency real-time processing required by the advanced algorithms necessitates calls for high computational performance. Also the physical size and thermal design power is restricted by cosmetic and comfort reasons.

On top of the strict quantitative characteristics, a programmable design is preferred. Tailored fixed function hardware solutions offer the best performance per area or power, but lack the capability to support future functionality improvements on the manufactured device by means of software updates. DSP-like processors augmented with SIMD instructions is a design space with high performance low power combined with flexibility of programmability. Typical DSP designs have static low power multi-issue datapaths, which, when combined with SIMD instruction sets increase the power-performance by reducing the number of decoded instruction bits per executed operation. Such processors are widely available "off-the-shelf". However, they are designed to support a wide range of algorithms, thus not receiving the best power performance if the design was tailored for a narrower set of applications.

In this paper we propose a *application-specific instruction-set processor (ASIP)* design tailored for advanced hearing aid algorithms. The strict requirements are reached with a careful selection of the set of SIMD and scalar function units and other data path components such as memory gather instructions and special instructions for optimized CORDIC processing. Further improvements in power-performance are received via

its use of the *transport-triggered architecture (TTA)* which alleviates the register file pressure, a typical bottleneck in DSP designs [3]. As a case study, the binaural speaker localization by [2] was implemented to evaluate the design.

The remainder of this paper is organized as follows. First, Section II reviews the related work on processor-based approaches for audio signal processing in hearing aid devices. Section III presents the implemented probabilistic binaural speaker localization algorithm. Section IV guides through the used TTA design flow and the design choices for the proposed processor architecture which is evaluated in Section V. The paper is concluded in Section VI.

II. Related Work

To the best of our knowledge, this is the first publication of a realistic programmable processing architecture for the probabilistic binaural speaker localization or a comparable computationally complex algorithm for hearing aid devices. However, to provide an overview of other state-of-the-art hearing aid processors, we revise a set of architectures for complex modulated filtering [4] and noise reduction [5] algorithms. These algorithms are approximately $15\times$ less computationally demanding than the ones supported by the proposed design.

A customized Tensilica/Cadence Xtensa LX4 32-bit RISC processor for digital hearing aid systems is presented in [6]. It includes instruction set extensions for custom instructions and register files and variable length instruction encoding. Real-time computation consumes about 2 mW at 13.24 MHz on a silicon area of 0.623 mm^2 (TSMC40LP).

A customized ASIP from the RAPANUI project [7] can perform real-time processing with 0.18 mW at 2.18 MHz, and a silicon area of $0.14mm^2$ (TSMC40LP). The design was extended in [8] to a generic VLIW-SIMD ASIP for both previously mentioned audio signal processing algorithms with complex multiplication and count leading zeros instructions.

In [9] a Silicon Hive Pearl VLIW-ASIP was proposed for beamforming, feedback cancellation, FIR filter bank, compression and noise reduction. The design had a 3-issue-slot 16-bit VLIW instruction format, 40-bit registers for immediate results from the 32-bit datapath, fixed-point arithmetic units and custom instructions. Clock-frequency of 11 MHz achieved real-time computation on a silicon area of 0.49 mm^2 (TSMC C65G) consuming 0.964 mW after voltage and frequency scaling.

A low power fixed-point DSP for FFT was proposed in [10] via optimized instruction schedules and bit-reversed addressing. In addition to 16/32-bit datapath and 16-bit multipliers, it included up to 128-bit wide SIMD instructions. At a clock frequency of 8 MHz the design required less than 40% of the available time for real-time computation of FFT-based filterbank, compression and feedback cancellation.

III. Probabilistic Binaural Speaker Localization

Figure 1 illustrates the method for binaural speaker localization used in the proposed processor design. Given the

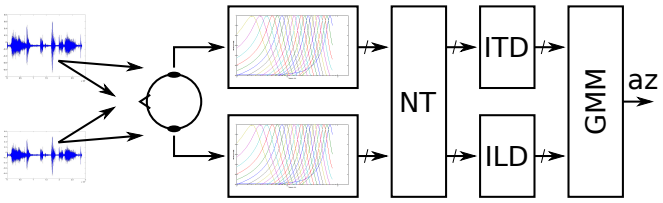

Fig. 1. Schematic diagram of the propabilistic binaural speaker localization algorithm. First the stereo microphone input from the stereo hearing aid device is preprocessed in a gammatone filter bank succeeded by neural transduction (NT). Interaural time (ITD) and level differences (ILD) are later classified by an azimuth-dependent pretrained gaussian mixture model (GMM).

velocity of sound, phase delays, named *interaural time differences* (ITD), between both ears or microphones can be used to estimate the position of an acoustic source in the horizontal plane of a binaural auditory system [11]. Especially at higher frequencies, when the wavelength becomes smaller than the diameter of the head and leads to ambiguous ITD informations, *interaural level differences* (ILD) can contribute to the localization process. [12] simulates the human auditory system by an auditory front-end consisting of a fourth-order, 32-channel, non-linear *gammatone filterbank* (GTFB). Inner hair cell-processing is simulated through neural transduction (NT), square-root compressing the half wave rectified audio cues. The succeeding interaural level and time difference calculations constitute a multidimensional feature space, which is finally classified by an azimuth-dependent pre-trained gaussian mixture model (GMM).

IV. Processor design

Processor designs for hearing aid devices target highest power efficiency after reaching the real-time constraint with the auxiliary constraint of a small silicon footprint, in order to enable durable and small hearing aid devices. In the proposed implementation, the following design choices contributed towards fulfilling the requirements.

Algorithmic improvements help reducing the required computational effort with a trade-off in accuracy/quality. As the ITD-calculation necessitates only the logarithmic cross-correlation values and the maximum position of the cross-correlation function is equal in the logarithmic scale, due to the properties of continuous and steadily growing nature of the logarithmic function, only the logarithmic cross-correlation value needs to be calculated. This eliminates the division operation for the normalization of the cross-correlation and replaces it with the subtraction of the logarithms of the nominator and denominator. In addition, the logarithm of the denominator transforms the square-root calculation of the mean-free signal energies to a multiplication with the factor $\frac{1}{2}$ (that can be implemented as a right shift of one) of the logarithm and thereby eliminates one expensive CORDIC calculation. The parallel calculation of the necessary mean values of the auditory frontend output is integrated between the half-wave rectification and square root compression in the

neural transduction processing. For the denominator, the sum of these squared samples is needed and represents more than 50 % of the total computational complexity of the ITD, if calculated for every displacement. However, as these sums have no dependencies between both channels, the squared sample values are calculated concurrently and saved by integrating all previous values of the same cue, with the trade-off of 8-bit loss in accuracy, thus reducing the calculation of the denominator to maximum of two memory accesses.

Utilizing **fixed-point arithmetic** in the entire application in comparison to floating-point reduces the required silicon area and the overall computation latency. The whole algorithm was simulated with different fixed-point formats and an appropriate format was carefully chosen for every algorithmic stage in terms of value coverage and accuracy. A satisfactory speaker localization can be granted with the use of 32-bit integers and different fractional parts throughout the implementation, retaining the correct localization of the speaker.

Exploiting **data-level parallelism** reduces the amount of instruction stream overheads, as *single instruction multiple data* (SIMD) instructions perform the same operation on multiple data with a single opcode and operand register instruction fields. The schematic of the binaural speaker localization in Figure 1 gives an overview on possible points of parallelization. As the algorithm operates on stereo data, processing both channels using 64-bit wide vectors of two elements would be a natural parallelization point. However, in order to maximize the benefits available from using SIMD operations we exploited the data-level parallelism within each separate audio channel for the 32 different gammatone channels treated with identical operations, thus resulting in 1024-bit vector operations utilized throughout the whole algorithm.

In order to maximize the flexibility of the design and the interoperability with a variety of different audio processing related algorithms, we included a set of common basic SIMD operations operating on 32 x 32b integer vectors (abs, add, and, eq, gt, lt, max, min, or, shl, shr, sub, and xor). We also included vector-scalar interoperability operations for element extract, insert and broadcast and a conditional element select.

Custom operations (special instructions) offer high energy efficiency and computational performance with the cost of additional silicon area and reduced generality. Considering the design goal of flexibility, we avoided using application specific operations to cover significant parts of the implementation, but preferred operations useful for wider range of hearing aid related audio signal processing.

Multiple algorithmic stages throughout the algorithm use mathematically complex operations on fixed point values, including square root (NT), logarithm (ITD, ILD, GMM) and exponential (GMM) calculation, which are calculated using CORDIC iterations. An element-wise conditional add subtract instruction accelerates the CORDIC core loop for all operational modes and mathematic operators. In order to further accelerate the preprocessing, a count leading zeros instruction was used to normalize the input values. To accelerate the implemented fixed point arithmetic, particularly in the gammatone filter stage, a combined multiply shift operation was added to the SIMD instruction set. The alignment of the auditory cues by the group delay of each gammatone filter was realized by gathering the input data from the input queue. Thus, a custom gather load instruction was added, sequentially loading 32-bit elements to a 32-element vector from an address specified by a constant base address and a gammatone filter dependent offset.

Using the **Transport-Triggered Architecture** (TTA) as a programming model reduces the processing latency by means of software register file bypassing and scalable instruction level parallelism. TTAs are programmed by defining data transfers in the processor interconnection (IC) network, which can be utilized to reduce the register file complexity and area in comparison to VLIW-based approaches. However, designing a compiler-supported TTA from the scratch involves high development effort, since it necessitates not only hardware design and verification, but also the engineering of the supporting software tools. To this end, *TTA-Based Co-Design Environment* (TCE) [13], a mature TTA design toolset that includes a retargetable C compiler, was utilized to design and program the TTA, and to generate the RTL implementation for evaluation purposes.

The designed TTA is shown in Figure 2, illustrating also the programmer visible connections in the IC. The utilization of the function units (FU) and the interconnection network was maximized by splitting the 32 SIMD registers to two equally sized 16-entry 1024-bit register files each having one write port and either one or two read ports. A direct connection was added from the SIMD ALU output to its inputs to enable moving wide results to the next wide operations maximizing the benefits of the TTA-specific software bypassing optimization. The interconnection network of the remaining four transport buses was manually pruned after reaching the real time design constraint which helped reaching an instruction width of 64 bits.

Unique in comparison to state-of-the-art hearing aid algorithms, the probabilistic binaural speaker localization requires a large constant memory for the parameters of the gaussian mixture model of 355.200 B (32 gammatone channels, with 15 gaussian components and 37 azimuth bins, 5 values each), which is realized by connecting an on package data memory (e.g. eDRAM) via a dedicated load-store unit. The parameters are thereby stored consecutively in the required order for calculation, thus the latency of the memory can be hidden by preloading memory in local registers in a pipelined fashion.

V. Evaluation

The maximum clock cycle count is defined by the sample frequency of the peripheral audio codec of 16 kHz and the chosen clock frequency of $50\ MHz$, representing a common tradeoff between power consumption and computing performance. Accordingly, 800 000 clock cycles are available for the processing of the binaural audio cue with 512 stereo samples. The proposed design requires 751 549 cycles for each audio

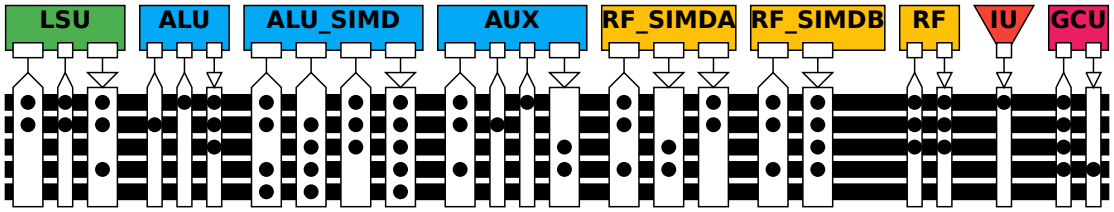

Fig. 2. Datapath of the proposed TTA processor for binaural speaker localization.

frame, thus satisfying the real time processing constraint. The remaining cycles can be used to reduce the processor clock frequency to 47 MHz, hence potentially increasing the energy efficiency.

A 28 nm low power process was used for the synthesis of the proposed design in combination with an on chip instruction memory (11-bit addressing of 64-bit data words). The combined area requirement of the core and the instruction memory is 300985 μm^2 of global cell area, with the processor core taking 88.6 % of the total area and the rest spent for the instruction memory. Beside the load-independent area estimation, the typical processing for one audio frame has been simulated on signal level and fed in the power estimation tool for the final architecture. With this load the processor shows an average power consumption of 11.9 mW, of which 92.2 % is consumed by the datapath components and by the instruction memory at a global operating voltage of 1 V. An 32 nm 400 kB eDRAM data memory is estimated to 0.85 mm^2 and a power consumption of 1.7 mW with an utilization of 4.6%.

Of the datapath components, the `alu_simd` consumes 44.1 % of the total computational area and 76.4 % of the power (excluding instruction memory) at an average utilization of 42.1% of the cycles. As most of the power is spent in computation related components, an excellent power efficiency is indicated. The overheads incurred by the programmability are very low, which can be only reached with a very power efficient streamlined control logic like TTAs have. The rest of the function units neccessitate only 4.4 % and 2.0 % of the processing related area and power in total.

The vector register files take a large fraction of the chip area (43.2 %), but consume very little power (10.5 %) thanks to the TTA's software bypassing feature which allows moving results between function units without touching the register file. The SIMD register file utilization of approximately 45 % and the corresponding speed-up of the algorithm justify the large area commitment. The interconnection network shows an average utilization across all transport buses of 43 %, while consuming 5.9 % and 10% of the computational related total processor area and power, which is a good tradeoff.

VI. Conclusion

This paper proposed an energy efficient programmable processor design for binaural speaker localization algorithms and other high computational complexity audio processing algorithms in the context of hearing aid devices.

The stringent requirements were reached by utilizing a 32-element wide SIMD datapath, optimizing the memory accesses, and adding a few carefully chosen application specific special instructions. The power efficiency of the wide-SIMD datapath was improved with the TTA model which enabled register file related area and energy savings. To the best of ours knowledge, this is the first implementation of the probabilistic binaural speaker localization for hearing aid devices, with a power consumption as low as 11.9 mW at 50 MHz clock frequency and a silicon area estimate of 0.3 mm^2 for a 28 nm standard CMOS technology, fitting well in the physical limits of a hearing aid device.

References

[1] WHO, "Deafness and hearing loss," March 2015, available: http://www.who.int/mediacentre/factsheets/fs300/en/ [Accessed: 25 Feb 2016].

[2] T. May, "Binaural scene analysis localization, detection and recognition of speakers in complex acoustic scenes," Ph.D. dissertation, Technische Universiteit Eindhoven, Eindhoven, Netherlands, 2012.

[3] J. Hoogerbrugge and H. Corporaal, "Register file port requirements of transport triggered architectures," in *Proceedings of the 27th Annual International Symposium on Microarchitecture*, ser. MICRO 27. New York, NY, USA: ACM, 1994, pp. 191–195. [Online]. Available: http://doi.acm.org/10.1145/192724.192751

[4] O. Semiconductor, "Wola filterbank coprocessor: Introductory concepts and techniques," April 2009, available: http://www.onsemi.com/pub/Collateral/AND8382-D.PDF [Accessed: 25 Feb 2016].

[5] N. Westerlund, M. Dahl, and I. Claesson, "Speech Enhancement for Personal Communication Using an Adaptive Gain Equalizer," *Signal Processing*, vol. 85, no. 6, pp. 1089–1101, 2005.

[6] N. Werner, G. Payá-Vayá, and H. Blume, "Case Study: Using the Xtensa LX4 Configurable Processor for Hearing Aid Applications," in *Proceedings of the ICT.OPEN 2013*. ICT, 2013, pp. 27–32.

[7] G. Payá-Vayá, "Design and Analysis of a Generic VLIW Processor for Multimedia Applications," Ph.D. dissertation, Institute of Microelectronic Systems, Leibniz Universität Hannover, 2011.

[8] J. Hartig, L. Gerlach, G. Pay-Vay, and H. Blume, "Customizing a vliw-simd application-specific instruction-set processor for hearing aid devices," in *Proceedings of the 2014 International Workshop on Signal Processing Systems*, ser. SIPS '14. IEEE, 2014, pp. 1–6.

[9] P. Qiao, H. Corporaal, and M. Lindwer, "A 0.964 mW Digital Hearing Aid System," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2011, pp. 1–4.

[10] Y. Ku *et al.*, "A High Performance Hearing Aid System with Fully Programmable Ultra Low Power DSP," in *Consumer Electronics (ICCE), Int. Conf. on*. IEEE, 2013, pp. 352–353.

[11] T. May *et al.*, "A probabilistic model for robust acoustic localization based on an auditory front-end," in *Proceedings of the NAG/DAGA*, Rotterdam, Netherlands, 2009, p. 254.

[12] ——, "A probabilistic model for robust localization based on a binaural auditory front-end," vol. 19, no. 1, pp. 1–13, 2011.

[13] O. Esko, P. Jääskeläinen, P. Huerta, C. S. de La Lama, J. Takala, and J. I. Martinez, "Customized exposed datapath soft-core design flow with compiler support," in *Proceedings of the 2010 International Conference on Field Programmable Logic and Applications*, ser. FPL '10. IEEE, 2010, pp. 217–222.