IET Circuits, Devices & Systems

Research Article

IET Journals
The Institution of Engineering and Technology

High-performance DSP platform for digital hearing aid SoC with flexible noise estimation

ISSN 1751-858X
Received on 22nd December 2017
Revised 31st July 2018
Accepted on 4th March 2019
E-First on 1st July 2019
doi: 10.1049/iet-cds.2018.5374
www.ietdl.org

Sang-Won Kim[1], Min-Joon Kim[1], Jae-Seok Kim[1] ✉

[1]*School of Electrical and Electronic Engineering, Yonsei University, Seoul, Republic of Korea*

✉ *E-mail: jaekim@yonsei.ac.kr*

Abstract: Flexibility and programmability of hearing aids are important because the algorithms applied to hearing aids should be changeable based on different types of hearing impairment and the ambient environment of the user. This paper proposes a high-performance digital signal processing (DSP) platform for a digital hearing aid system on a chip (SoC) with flexible noise estimation. The proposed DSP platform comprises several dedicated accelerators and an application-specific instruction-set processor (ASIP) to achieve flexibility. To handle complex hearing aid algorithms in real time, the main algorithms of hearing aids are executed by hardware accelerators and only environment-sensitive parts of the applied algorithms are implemented as the ASIP. Simulation results show that the proposed DSP platform can handle complex and high-performance algorithms in real time, and that it provides better quality in terms of noise handling by adapting the noise estimation algorithms suitable for the noise environment. The chip area of authors' DSP design is 2.71 mm^2, and it consumes 1.3 mW at 1 V operation, 8 MHz clock frequency with a 65 nm high threshold voltage (HVT) standard cell library.

1 Introduction

Many modern hearing aids employ several algorithms, such as noise reduction (NR), wide dynamic range compression (WDRC), and feedback cancellation (FBC). Flexibility and programmability of the hearing aids are important because the algorithms applied to the hearing aids should be changeable based on the different types of hearing impairment and the ambient environment of the user [1]. Power efficiency and processing speed are also important issues in hearing aids because of constraints such as limited power consumption budget and real-time operation. For low power consumption, application-specific integrated circuits (ASIC) are traditionally used for hearing aids designs [2]. However, modifying the algorithms on ASICs requires very high cost and long time. A general purpose digital signal processor (GPDSP) platform is suggested to solve this problem [3]. This platform provides good flexibility, but its power consumption is extremely high. In order to make full use of ASIC power efficiency, an application-specific instruction set processor (ASIP) is another popular choice. This is because ASIP is more power efficient than GPDSP and provides good flexibility. In previous experiments, ASIP was used based on reduced instruction set computer (RISC) to efficiently execute hearing aid algorithms [4]. The customisation is usually performed by inserting complex custom instructions into the base instruction-set of the generic processor. Then, not only the RISC instruction set but also the module level flexibility in implementation with the ASIP is suggested. This combines the ASIC and ASIP with an interrupt scheme [5]. Whenever a user-defined module on the ASIP replaces some hardware accelerators, a module level clock-gating scheme can be used to shut down the clock source and the interrupt scheme works.

This paper proposes a high-performance DSP platform for a hearing aid SoC, which has an optimised structure for user customisation. The DSP platform comprises an ASIP and several hardware accelerators for flexibility in design. Although the interconnection between hardware accelerators and ASIP shows excellent flexibility at the module level [5], it is less focused on optimisation for real-time operation and efficient implementation. If modules implemented as ASIP are too many and complicated, real-time operation could be impossible and power consumption could be inefficient, because ASIP is relatively slower and more power consuming than ASIC. If modules implemented as ASIP are not specified and fixed, the same module could be implemented as both a hardware accelerator and ASIP. This is inefficient in terms of design cost and power consumption, even if clock-gating is used. Along with flexibility, it is important to discuss these points because of the nature of hearing aids that must operate with low power and low latency. The modules implemented in ASIP should be minimised and specified. Therefore, only the noise estimation algorithm and WDRC gain controller are implemented as ASIP in the proposed design. This is because they are environment-sensitive parts that, according to the user, need to be changeable. By reducing the modules implemented as ASIP, our design obtains enough time for real-time operation of more complex noise estimation algorithms and better performance in noise handling.

The accelerators are implemented as an ASIC version for power efficiency and they execute main hearing aid algorithms, which are FFT/IFFT filter banks, NR, WDRC, and FBC. The 128-point module sharing FFT/IFFT filter banks are used to obtain a high-frequency resolution for high-performance hearing aids. This paper is the first to implement the hearing aid SoC using 128-point FFT/IFFT for high performance, which make it possible to analyse the speech signal on assuming its' stationarity. The NR accelerator uses the structure of multiband spectral subtraction (*mband*) [6] of which the noise estimation algorithm can be modified by ASIP. This is because the mband provides an excellent trade-off between computational complexity and noise suppressing performance. The simulation results show that our design can handle several noise estimation algorithms in real-time and the performance is improved by changing the noise estimation algorithm proper to different noise environments.

The paper is organised as follows. Section 2 presents an overview of the proposed DSP platform. Section 3 explains hardware accelerators. In Section 4, the ASIP and performance of noise estimation algorithms are shown. Implementation results are presented in Section 5. Finally, conclusions are drawn in Section 6.

2 Proposed DSP platform

Our design is focused on achieving high performance and user customisation under the constraints of hearing aids, such as low-power consumption and real-time operation. The hardware block diagram and operational flow of the DSP platform proposed here are illustrated in Figs. 1 and 2, respectively. The 32-bit DSP

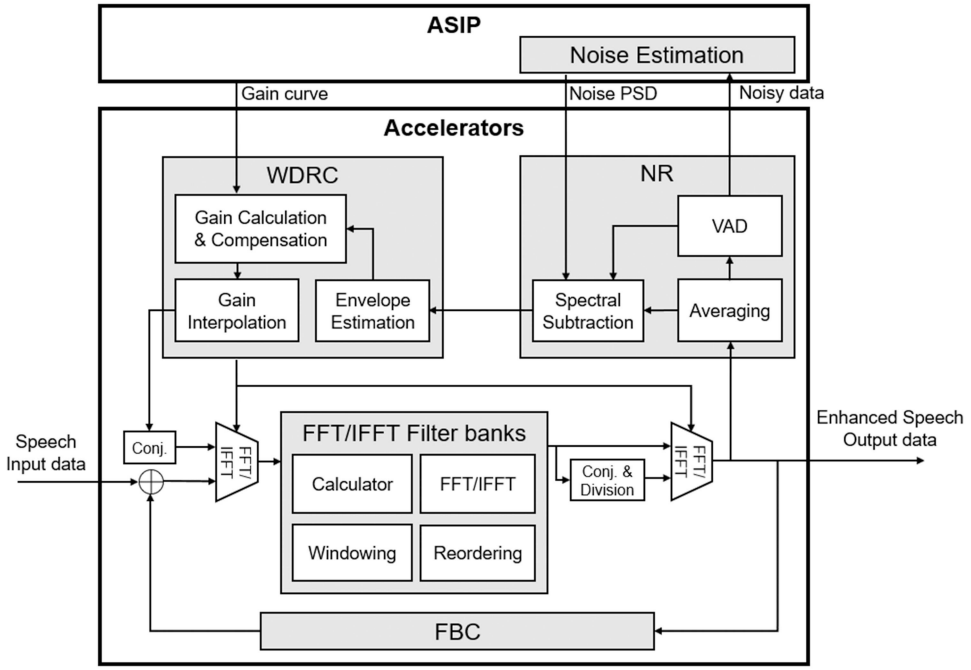

Fig. 1 *Hardware block diagram of the DSP platform*

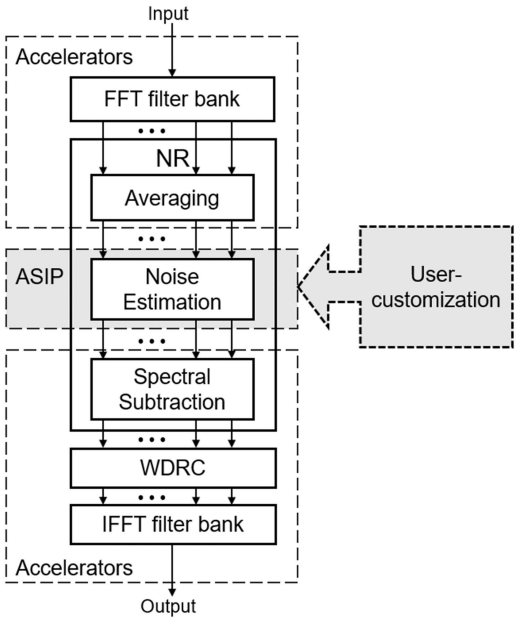

Fig. 2 *Operational flow of the DSP platform*

platform consists of several hardware accelerators and an ASIP. The 16-bit 16 kHZ samples are processed on the DSP platform and the number of bits for ASIP is 24. The number of bits in each data is determined by fixed point simulation, which are partly shown in Fig. 3. The number of bits for data is set big enough to take full advantage of high FFT point.

The accelerators are module sharing 128-point FFT/IFFT filter banks, NR, WDRC, and FBC. They are implemented as ASIC for fast and low power handling of the fixed and intense computing task for hearing aids. The module sharing 128-point FFT/IFFT filter banks work as an analysis filter bank (AFB) and synthesis filter bank (SFB) for high-frequency resolution. They decompose the input speech into frames and recombine the components overlapping after the NR and WDRC algorithms are performed sequentially on each sub-band in the frequency domain. The NR accelerator removes noise with NR algorithms to improve speech quality and intelligibility. The WDRC accelerator maps the input speech signals to the appropriate level of output according to a user-set gain I/O curve to facilitate easier hearing. The ASIP executes user customised noise power estimation algorithms and

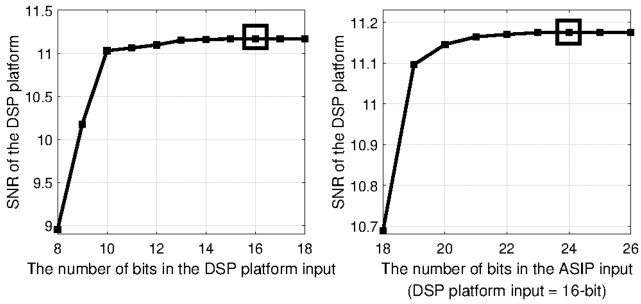

Fig. 3 *SNR of DSP platform by the number of bits in data*

controls the WDRC gain compensation level. The hardware accelerators and ASIP share data memory for operations such as noise estimation of NR and gain compensation of WDRC.

3 Hardware accelerators

3.1 Module sharing 128-point FFT/IFFT filter banks

The existing results about hearing aids implementation used FFT point smaller than 128 [1, 5]. However, this is not enough for high-performance speech signal processing. Processing the speech signal, quasi-stationarity should be considered. For this, frame size is generally set 20–40 ms, 320–640 samples in 16 kHz of sampling rate, as possible interval for analysis assuming stationarity [7]. Also, recent papers about spectral subtraction use 256-point or more for FFT [8–10]. From that reason, we decided that higher point FFT is needed for higher performance. The 128-point is a compromise between performance and latency.

Our design of the module sharing 128-point FFT/IFFT filter banks uses a pipelined radix-2 single-path delay feedback (R2SDF) architecture for the real-time processing. These accelerators work as AFB and SFB to decompose and reconstruct the speech signals and provide a high-frequency resolution for hearing aids. The module sharing 128-point FFT filter bank and IFFT filter bank share most of their structure for a better hardware utilisation efficiency. Fig. 4 shows the block diagram of our FFT/IFFT filter banks. These filter banks comprise an input buffer, windowing block, FFT/IFFT block, and reordering block. A calculator block is designed for 2N-point FFT/IFFT, which is a method using the fact that the imaginary part of real sound is zero. It makes a 2N-point FFT possible using N-point FFT and auxiliary calculation as shown below

$$\begin{cases} x_1(n) = g(2n), \\ x_2(n) = g(2n+1), \qquad 0 \le n \le N-1 \\ x(n) = x_1(n) + jx_2(n) \end{cases} \tag{1}$$

where $g(n)$ is real-valued sequence of 2N-point. FFT of $g(n)$ and IFFT of $G(k)$ are calculated using the relationship between $G(k)$ and $X(k)$ as below

$$G(k) = X(k)A(k) + X^*(N-k)B(k), \quad k = 0, 1, \ldots, N-1 \tag{2}$$

$$X(k) = G(k)A^*(k) + G^*(N-k)B^*(k), \quad k = 0, 1, \ldots, N-1 \tag{3}$$

where $A(k)$ and $B(k)$ are the twiddle factors saved in memory and $X(k)$ is the DFT result of $x(n)$. The 2N-point FFT is suitable for the module sharing FFT/IFFT architecture because (2) and (3) have a very similar format. It can save power and implementation area, by 19 and 12%, respectively. The cycle count is also decreased by 27%. The filter banks only make a 64-point delay, because of the 2N-point, which is 4 ms in case of 50% overlap-add and 16 kHz sampling frequency. It is a tolerable delay for the real-time processing in hearing aids [11].

3.2 NR accelerator

A modified NR algorithm based on multiband spectral subtraction [6] is designed. It is one of the well-known spectral subtraction algorithms and has an excellent trade-off between computational complexity and NR performance. However, conventional multiband spectral subtraction is not suitable for hearing aids because of their power and latency problems [12]. We developed an optimised version of it for user customisation and efficient hardware implementation, of which the block diagram is illustrated in Fig. 5. Among the components of the NR algorithm, only the noise power estimation algorithm is implemented in the ASIP to obtain design flexibility. This is because it is environment-sensitive and should vary according to different hearing impairments [12, 13]. Other parts of the NR are implemented as an ASIC version for low latency and power efficiency. They exchange data with the noise estimation module in the ASIP using shared memory. It can provide enough flexibility for different kinds of hearing impairments and noise condition changes while being more power efficient than implementing all NR algorithms as an ASIP. The NR algorithm works as below

$$\left|\hat{S}_i(k)\right|^2 = \left|Y_i(k)\right|^2 - \alpha_i\delta_i\left|\hat{D}_i(k)\right|^2 \quad f_i \le k \le l_i \tag{4}$$

where $S_i(k)$ and $D_i(k)$ are the magnitude spectra of the clean speech and the noise of the ith subband, respectively. $Y_i(k)$ is smoothed through averaging between several frequency bands and frequency bins. f_i and l_i are the first and last frequency bins of the ith frequency band, respectively. α_i is the over-subtraction factor and δ_i is a tweaking factor of band i that can be individually set for each frequency band to customise the noise removal properties. The band-specific over-subtraction factor α_i is a function of the segmental SNR_i of the ith frequency band. The value setting is similar to that of mband but is modified slightly to suit hardware implementation, which is described as follows

$$SNR_i(\text{dB}) = 10\log_{10}\frac{\sum_{m=-\infty}^{\infty}|Y_i(k)|^2}{\sum_{m=-\infty}^{\infty}\left|\hat{D}_i(k)\right|^2} \tag{5}$$

$$\alpha_i = \begin{cases} 5 & SNR_i < -5 \\ 4 - \frac{1}{8}(SNR_i) & -5 \le SNR_i \le 20 \\ 1 & SNR_i > 20 \end{cases} \tag{6}$$

The tweaking factor δ_i is set to 1 in the first band and 2.5 in the other bands, since most speech energy is distributed in the lower frequencies and they need a smaller tweaking factors to minimise speech distortion. The negative values from the subtraction in (4) were floored to the noisy spectrum as

$$\left|\hat{y}(n,i)\right| = \begin{cases} \left|\hat{y}(n,i)\right|, & \text{if } \left|\hat{y}(n,i)\right| > \beta\left|a(n,i)\right| \\ \beta|a(n,i)|, & \text{else} \end{cases} \tag{7}$$

The coefficients for the calculations above are simplified for fixed point calculations. The multiplication of factors is replaced with shifting and addition. Non-linear calculations, including logarithm and exponential, are approximated using a look-up table method. The entropy voice activity detection (VAD) [14, 15] is used to enhance the accuracy.

3.3 WDRC accelerator

The WDRC accelerator calculates the gain of the input signals and controls the gain compensation based on the gain I/O curves, which are adapted for each subband. The block diagram of the WDRC and a typical example of a gain I/O curve are illustrated in Fig. 6. The gain envelope estimation is a process of calculating the input sound pressure level (SPL), and the gain interpolation is a smoothing process to reduce aliasing. In gain computation, the default gain I/O curve cuts off the input signal with SPL under −70 dB, because they are considered as noise. The gain I/O curves need

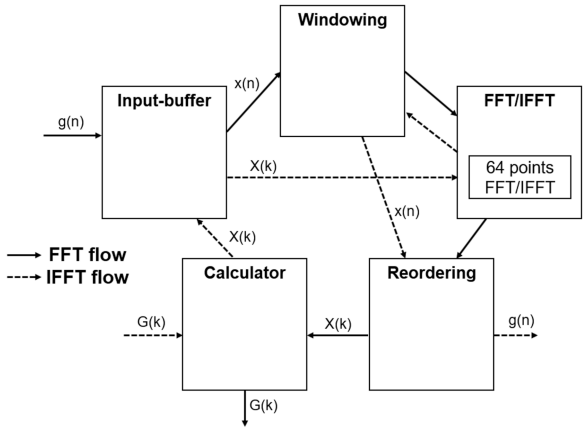

Fig. 4 *Block diagram of FFT/IFFT filter banks*

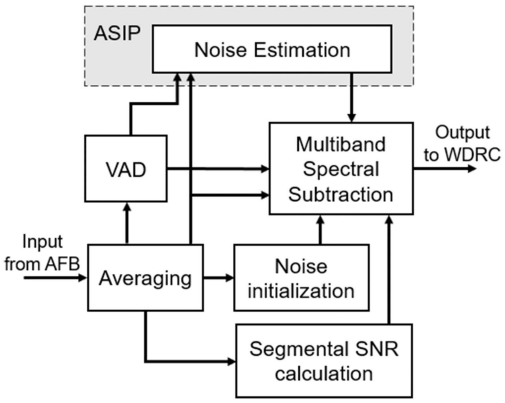

Fig. 5 *Block diagram of the NR accelerator*

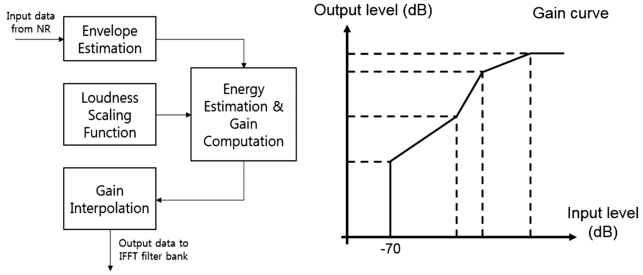

Fig. 6 *Block diagram of the WDRC accelerator and gain I/O curve*

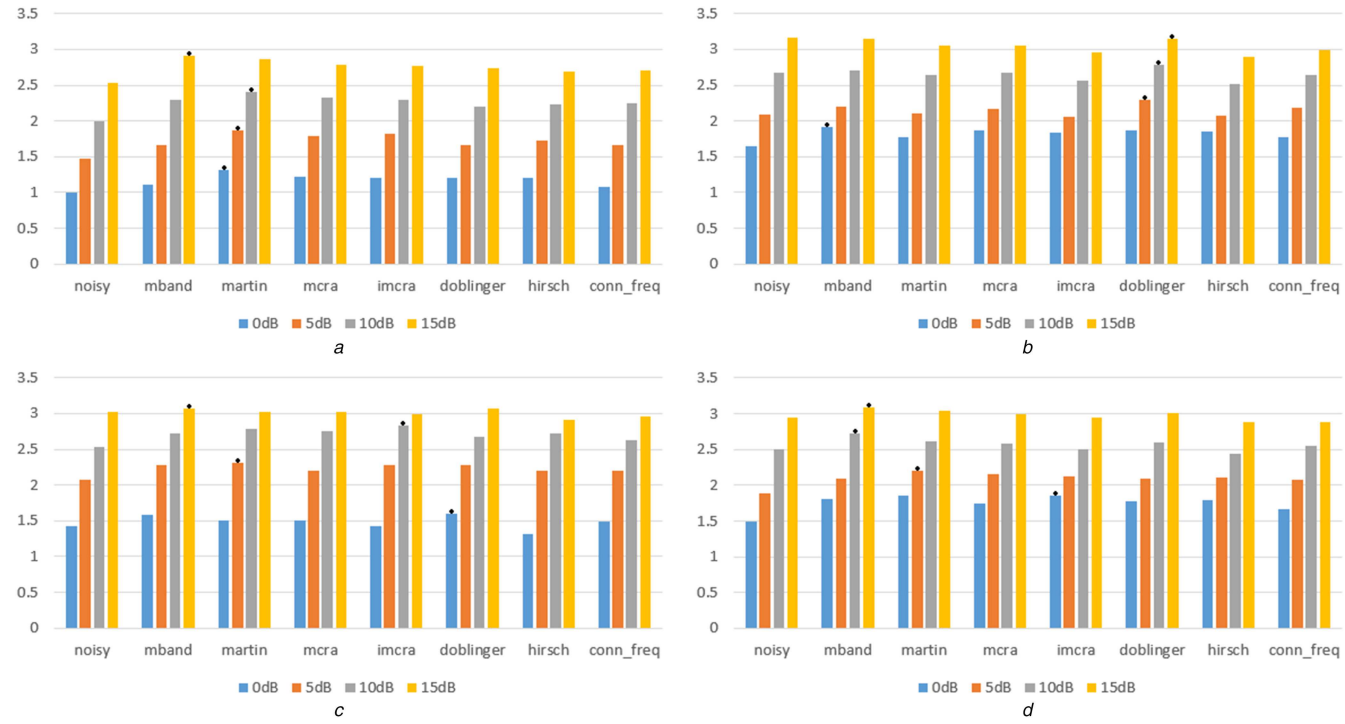

Fig. 7 *Comparison of OVL under*
(a) White, ***(b)*** Babble, ***(c)*** Factory, ***(d)*** Car noise condition

Table 1 Operation amount of FBC algorithm

Operation	Addition	Multiplication
SAF [17]	$M(3+6\log_2 M)$	$M(8+3\log_2 M)$
NLMS [18]	$M(3L-1)$	$M(3L-21)$
FDADF [19]	$M(4+18\log_2 M)$	$M(10+9\log_2 M)$
AP [20]	$M(2L+25P)$	$2L+24P-1$

to be flexible and programmable because they vary according to different types of hearing impairment of each user. To meet that requirement, the WDRC accelerator receives the gain I/O curve fitted for each user from the ASIP through shared memory.

3.4 FBC accelerator

FBC is used to suppress the re-amplification of output data caused by the short distance between the microphone and receiver. Acoustic feedback is uncomfortable and limits the maximum usable gain of hearing aids [16]. FBC in hearing aids includes estimating the feedback signals and subtracting it from the input speech signal. The subband adaptive filter (SAF) is used for the FBC of our design [17], which provides better quality than the normalised least mean squares (NLMS) algorithms [18], frequency-domain adaptive filter (FDADF) [19], and affine projection (AP) algorithms [20]. Table 1 shows the computational complexity of each FBC algorithms where M is half the frame length, L is the filter order, and P is the projection order of the AP algorithm. It shows that the SAF has the least operational amount.

4 Application-specific instruction-set processor

The main purpose of proposed DSP platform is obtaining flexibility of design from ASIP and maintain processing speed for real-time operation of complex algorithm in ASIP by using ASIC version hardware accelerators. Only ASIC part is designed, then FPGA Zynq and its DSP are used for ASIC-ASIP co-simulation. The input of ASIP is 24-bit speech power data as shown in Fig. 7. A 32-bit customised programmable architecture is most used to process the data for noise estimation algorithms. The microarchitecture of ASIP must be pipelined to take full advantage of the time resulting from the rapid processing speed of ASIC design. Memory data path size and instruction sets should be customised by the simulation with several noise estimation algorithms. The simulation results using Zynq and its DSP show that this system works in real time, which is shown in Table 2.

Table 2 Cycle counts of noise estimation algorithms

NE algorithms	Cycle count	Operation time, μs	Proportion in idle time, %
mband	2176	272	8.5
martin	20,712	2589	80.7
MCRA	15,768	1971	61.4
IMCRA	14,496	1812	56.5
doblinger	12,344	1543	48.1
hirsch	13,456	1682	52.4
conn_freq	23,680	2960	92.2

Fig. 7 illustrates the performance evaluation of our design with seven typical noise estimation algorithms [21] using composite measures under the babble, car, train, and white noise from the NOIZEUS database [22]. The composite measure was designed to reduce the uncertainty in a subjective test and provides the overall effect using the mean opinion score (OVL) scale, which has an index from one (bad) to five (excellent). The graph of the noise estimation algorithm, which has highest OVL score in each SNR is dotted on the top.

Fig. 7*a* shows the OVL scores of our design with several noise estimation algorithms under the white noise. All methods provide quality improvement because white noise is predictable and non-stationary. The martin has best performance in most SNR input levels except the 15 dB SNR input because it is designed for a highly non-stationary environment. The IMCRA also shows better quality improvement than the mband because of its high computational complexity and high performance.

In terms of non-stationary environments, such as babble, car, and train, the noise estimation algorithms cannot provide significant speech quality improvement because the non-stationary conditions degrade the performance of the VAD and noise estimation. Fig. 7*b* shows the OVL scores under the babble noise. Many algorithms have worse OVL scores that are even worse than noisy speech. The Doblinger algorithm provides the best OVL score for the most SNR input levels because it is properly designed for short time stationary noise environments. Figs. 7*c* and *d* show

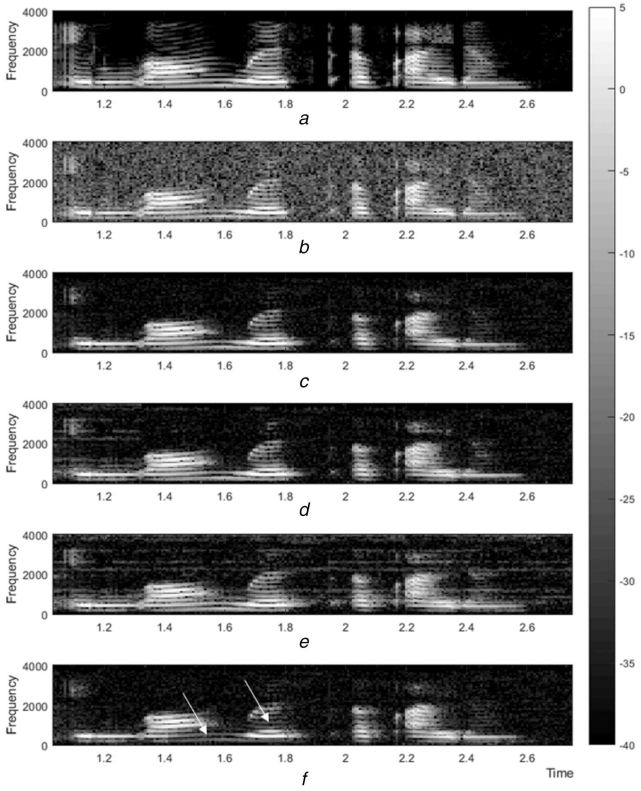

Fig. 8 *Spectrogram under the*
(a) Clean speech, ***(b)*** Speech with 15 dB white noise and the result processed by, ***(c)*** mband, ***(d)*** martin, ***(e)*** hirsch, ***(f)*** conn_freq

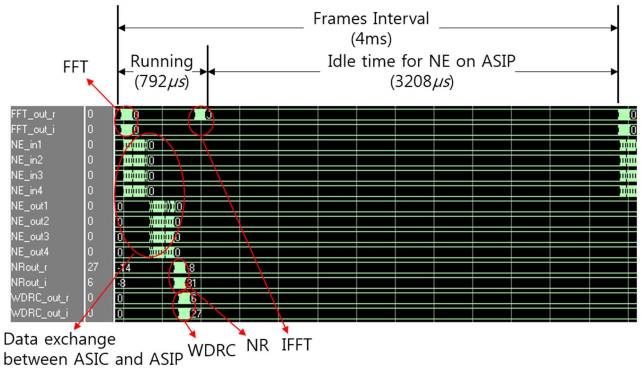

Fig. 9 *ASIC Simulation result*

Table 3 Comparison with previous work

	DATE 2011 [1]	ISCAS 2012 [4]	MJ 2015 [5]	This work
chip prototype	digital only	digital only	mixed-signal SoC	digital only
flexibility	ASIP only	ASIC + ASIP	ASIC + ASIP	ASIC + ASIP
hearing features	FBC, NR, BMF and WDRC	FBC, NR, and WDRC	FBC, NR, and WDRC	FBC, NR, and WDRC
DSP resolution	32	unknown	64	128
technology	65 nm	65 nm	130 nm	65 nm
supply	0.8–1.0 V	0.8 V	1.0–1.4 V	1.0 V
current	1.205 mA	1.625 mA	1.2 mA (DSP: 0.86 mA)	1.3 mA
size	0.49 mm^2	3.61 mm^2	9.5 mm^2	2.71 mm^2

the OVL scores of the car and train environments, respectively. Most methods used in the car conditions provide slightly better quality improvement than the train conditions. This is because the car noise condition is more stationary and most of the energy in a car noise is distributed at a low frequency.

Fig. 8 shows the spectrogram of the proposed design with various noise estimation algorithms presented in Fig. 7 under the 15 dB white noise condition. The spectrograms of mband and martin in Figs. 8*c* and *d* show that they have better noise suppressing performance than others under the white noise condition, which coincides with their high OVL scores. Otherwise the spectrogram of hirsch in Fig. 8*e* shows a lot of residual noise and the spectrogram of conn_freq shows many losses of speech components indicated by white arrows. These also coincide with their low OVL scores. The characteristics of each algorithm are shown in each spectrogram, but the specific characteristics of algorithms are beyond the scope of this paper. These spectrograms demonstrate that different noise estimation algorithms have significant performance differences in a particular noise environment, which is powerful reason for us to apply the flexibility using ASIP.

5 Implementation result and analysis

Implementing a DSP platform with only an ASIP renders the real-time processing of complex algorithms impossible because an ASIP is slow and power consuming. However, if the DSP platform is implemented with only an ASIC, design flexibility cannot be achieved. Therefore, our design reaches a compromise by using the ASIP and ASIC together. The hardware accelerators in the ASIC allows for sufficient operation time by executing iterative calculations quickly. The ASIP processes only noise estimation algorithms that should be changeable for each user, such that ample flexibility can be provided.

Fig. 9 shows a simulation result of implementing only ASIC parts of the DSP platform. The calculations that need to be processed by the ASIP are pre-computed and used at the test bench without delay. The simulation result shows that the inter-frame interval is 4 ms and the running time for the ASIC operation is 792 µs. This means that our system has an idle time of 3208 µs and can handle any other noise estimation algorithms for spectral subtraction [23, 24] in real time if the processing time of the ASIP is shorter than the idle time.

Table 2 shows the cycle counts of the noise estimation algorithms in Fig. 7, which can be operated in real time. The hearing aid DSP platform is fabricated in a 65 nm CMOS technology. The area of the DSP platform is 2.71 mm^2. The total current consumption of the DSP platform is only 1.3 mA with a 1 V voltage supply and 8 MHz operation clock frequency. Table 3 presents the comparison with previous works on hearing aids. The chip size of our design is 15% smaller than ISCAS 2012 [4] and 71% smaller than MJ 2015 [5]. Our design has the highest DSP resolution from the 128-point FFT/IFFT filter banks. The DSP in this work consumes only 1.3 mA, which is 20% less than ISCAS 2012 [4] that used the same technology.

6 Conclusion

This paper proposes a DSP platform for hearing aid SoC. The main focus of this work is to develop a flexible and high-performance DSP platform. Hardware accelerators and ASIP are used for this purpose. The ASICs quickly execute the iterative computing tasks that are not related to user-customisation. This provides sufficient idle time for the real-time processing of the complex algorithm in ASIP. The ASIP is used to separate the environment-sensitive parts from other parts for flexibility and open-programmability with low cost. Filter banks are designed to provide a high-frequency resolution for high performance, because most of the hearing aid signal processing is performed in the frequency domain. Comparison with references shows that this paper is the first to suggest which modules of hearing aids are to be placed in ASIC or ASIP, using both the ASIP and ASIC together. Our design is as flexible as others, but provides higher performance from higher

frequency resolution. With the approaches above, our design can handle complex and high-performance noise estimation algorithms in real-time, since ASICs can provide sufficient time. The area of our design is 2.71 mm^2 and consumes 1.3 mW (ASIC only) at 1 V operation, 8 MHz clock frequency with a 65 nm high threshold voltage (HVT) standard cell library.

7 Acknowledgment

This work was supported by the Industrial Core Technology Development Program (10052009, Development of Futuristic fusion-semiconductor core IP) funded by the Ministry of Trade, Industry & Energy, and was also supported by IDEC (IPC, EDA Tool, MPW).

8 References

[1] Qiao, P., Corporaal, H., Lindwer, M.: 'A 0.964 mW digital hearing aid system'. Design, Automation & Test in Europe, Grenoble, France, May 2011, pp. 1–4

[2] Wei, C.W., Kuo, Y.T., Chang, K.C., *et al.*: 'A low-power Mandarin-specific hearing aid chip'. Proc. IEEE Asian Solid-State Circuits Conf., Beijing, China, November 2010, pp. 333–336

[3] Chau, E., Sheikhzadeh, H., Brennan, R., *et al.*: 'A subband beamformer on an ultra low-power miniature DSP platform'. Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Orlando, FL, USA, May 2002, pp. 2953–2956

[4] Chang, K.C., Chen, Y.W., Kuo, Y.T., *et al.*: 'A low power hearing aid computing platform using lightweight processing elements'. Proc. IEEE Int. Symp. Circuits Systems, Seoul, Korea, May 2012, pp. 2785–2788

[5] Chen, L.M., Yu, Z.H., Chen, C.Y., *et al.*: 'A 1-V, 1.2-mA fully integrated SoC for digital hearing aids', *Microelectron. J.*, 2015, **46**, pp. 12–19

[6] Kamath, S., Loizou, P.: 'A multi-band spectral subtraction method for enhancing speech corrupted by colored noise'. Proc. IEEE Int. Conf. Acoustics Speech Signal Processing, Orlando, FL, USA, May 2002, p. 4164

[7] Thomas, F.Q.: '*Discrete-time speech signal processing: principles and practice*' (PrenticeHall, USA, 2001, 1st edn.)

[8] Upadhyaya, N., Karmakarb, A.: 'An improved multi-band spectral subtraction algorithm for enhancing speech in various noise environments'. Int. Conf. on Design and Manufacturing, Chennai, India, July 2013, pp. 312–321

[9] Miyazaki, R., Saruwatari, H., Inoue, T.: 'Musical-noise-free speech enhancement based on optimized iterative spectral subtraction', *IEEE Trans. Audio Speech Lang. Process.*, 2012, **20**, pp. 2080–2094

[10] Hu, H.T., Yu, C.: 'Adaptive noise spectral estimation for spectralsubtraction speech enhancement', *IET Signal Process.*, 2007, **1**, pp. 156–163

[11] Stone, M.A., Moore, B.C.J.: 'Tolerable hearing aid delays. I. Estimation of limits imposed by the auditory path alone using simulated hearing losses', *Ear Hear.*, 1999, **20**, pp. 182–191

[12] Wei, C.W., Tsai, C.C., FanJiang, Y., *et al.*: 'Analysis and implementation of low-power perceptual multiband noise reduction for the hearing aids application', *IET Circuits Devices Syst.*, 2014, **8**, pp. 516–525

[13] Singh, S., Tripathy, M., Anand, R.S.: 'Evaluation of noise estimation techniques for single-channel speech in low SNR noise environment'. Proc. IEEE Int. Conf. on Recent Advances and Innovations in Engineering, Jaipur, India, May 2014, pp. 1–5

[14] Jia, C., Xu, B.: 'An improved entropy-based endpoint detection algorithm'. Int. Symp. Chinese Spoken Language Processing, Taipei, Taiwan, ROC, August 2002, pp. 96–99

[15] Wu, B.F., Wang, K.C.: 'Robust endpoint detection based on the adaptive band partitioning spectral entropy in adverse environments', *IEEE Trans. Speech Audio Process.*, 2005, **13**, (5), pp. 762–775

[16] Siqueira, M.G., Alwan, A.: 'Steady-state analysis of continuous adaptation in acoustic feedback reduction systems for hearing-aids', *IEEE Trans. Speech Audio Process.*, 2000, **4**, pp. 443–453

[17] Farhang-Boroujeny, B., Wang, Z.: 'Adaptive filtering in subbands: design issues and experimental results for acoustic echo cancellation', *Signal Process.*, 1997, **61**, (3), pp. 213–223

[18] Haykin, S.: '*Adaptive filter theory*' (Prentice Hall, USA, 1996, 1st edn.)

[19] Sommen, P., VanGerwen, P., Kotmans, H., *et al.*: 'Convergence analysis of a frequency-domain adaptive filter with exponential power averaging and generalized window function', *IEEE Trans. Circuits Syst.*, 1987, **34**, pp. 788–798

[20] Lee, J., Park, Y.C., Youn, D.H.: 'Robust pseudo affine projection algorithm with variable step-size', *Electron. Lett.*, 2008, **44**, pp. 250–251

[21] Loizou, P.C.: '*Speech enhancement, theory and practice*' (CRC Press, USA, 2007)

[22] Available at http://ecs.utdallas.edu/loizou/speech/noizeus/, accessed July 2016

[23] Hu, H.T., Yu, C.: 'Adaptive noise spectral estimation for spectral subtraction speech enhancement', *IET Signal Process.*, 2007, **1**, (3), pp. 156–163

[24] Yamashita, K., Shimamura, T.: 'Nonstationary noise estimation using low-frequency regions for spectral subtraction', *IEEE Signal Process. Lett.*, 2005, **12**, (6), pp. 465–468