To appear in *ICSPAT'99 Proceedings*, November 1-4, 1999, Orlando, FL.

# An Ultra Low-power Programmable DSP System for Hearing Aids and Other Audio Applications

Todd Schneider, Robert Brennan, †Peter Balsiger and †Alexandre Heubi

Dspfactory Ltd.
611 Kumpf Drive, Unit 200
Waterloo, Ontario, N2V 1K8
Canada

†Institute of Microtechnology
University of Neuchâtel
Rue A.-L. Breguet 2, 2000 Neuchâtel,
Switzerland

## Abstract

This paper describes an application specific signal processor (ASSP) designed for miniature, ultra low-power, audio signal processing applications. The ASSP includes a high-fidelity weighted overlap-add (WOLA) filterbank, a 16-bit DSP core, two 14-bit A/D converters, a 14-bit D/A converter and a flexible set of peripherals. This design excels in applications that require efficient frequency-domain processing of one or two audio signals.

## 1. Introduction

For DSP designers, ultra low-power applications are extremely demanding. An example low-power application is a digital hearing aid. These devices must (i) consume less than 1 mW at 1 volt; (ii) fit into the ear canal (system size less than 6.5 x 3.5 x 2.5 mm); and (iii) provide sufficient flexibility and computational power to implement dynamic range compression, noise reduction, directional processing and other similar algorithms

Raw MIPS is a poor measure of performance in such applications. One must consider the power consumed to realize a particular algorithm [3]. Typically this figure of merit is expressed in mW/MIPS.

All of these features – miniature size, ultra low power consumption, targeted processing capability and flexibility make the design suitable for a wide range of portable audio and sensor signal processing applications.

## 2. Design Overview

Our design evolved from the realization that a number of signal processing algorithms utilize filtering as an underlying component. Dynamic range compression, noise reduction, directional processing and many other algorithms can be cast into a filtering framework. Thus, the basis of the design is efficient (yet flexible) filtering.

A significant increase in efficiency is achieved by "coding" filtering in hardware. A 16-bit DSP core controls this filtering engine and provides the flexibility needed to support a range of algorithms. A number of the filtering parameters are adjustable; thus, the filtering can be easily configured to meet the needs of many algorithms.

A parallel processing approach is employed: many DSP algorithms employ high-speed processing that operates at the incoming sample rate and lower-speed processing that operates at a frame rate or with slower time constants. For improved efficiency, the high-speed data processing is done in dedicated hardware. Lower-speed processing is implemented on the DSP core.

Compared to a fully software-programmable system, the architecture (Figure 1) trades some flexibility to achieve reduced power consumption. However, in ultra low-power applications, this trade-off is wise because of the reduced power consumption that can be realized. Hard coding part of the design also reduces memory requirements com-

pared to that of a fully programmable design – this further reduces chip size and power consumption. Finally, to achieve the lowest power operation, our design operates at 1 volt.

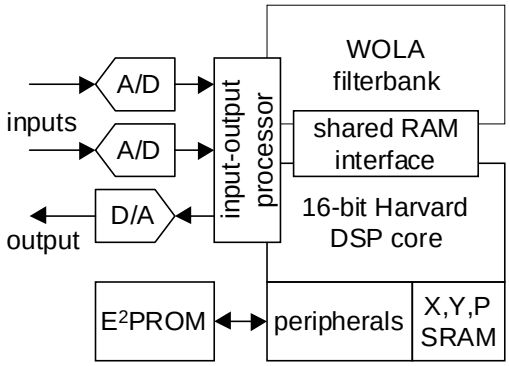


**Figure 1. Block diagram of DSP system**

The system is implemented on two ASICs. A digital chip on 0.18 μm CMOS contains the DSP core, RAM and WOLA filterbank The mixed-signal portions of the design are implemented on 1 μm CMOS. A separate, off-the-shelf, $E^2$PROM is used for non-volatile storage.

## 3. WOLA Filterbank

The filtering specifications for the target real-time audio processing applications are strict. Specifically, (i) less than 10 ms group delay, (ii) an adjustable number of bands, and (iii) at least 50 dB of gain adjustment in each band are required.

A frequency-domain filterbank was selected because it offers flexibility and simplified gain adjustment. Research on these filterbanks has primarily been done for speech coding applications. In speech coding, bitrate reduction is the goal; thus, the majority of these filterbanks use critical sampling and incorporate alias cancellation to maintain fidelity.

Simulations showed that large band gain adjustments (> 20 dB) in a critically sampled filterbank resulted in severe aliasing distortion because aliasing cancellation conditions [8] are violated. Distortion occurs because imaging in adjacent bands no longer cancels when the gain is adjusted. By oversampling, the aliased images are placed further away from gain adjusted bands so that they can be rejected by the synthesis filter.

Figure 2 and Figure 3 show the design of the analysis and synthesis sections of the WOLA filterbank [1],[2].

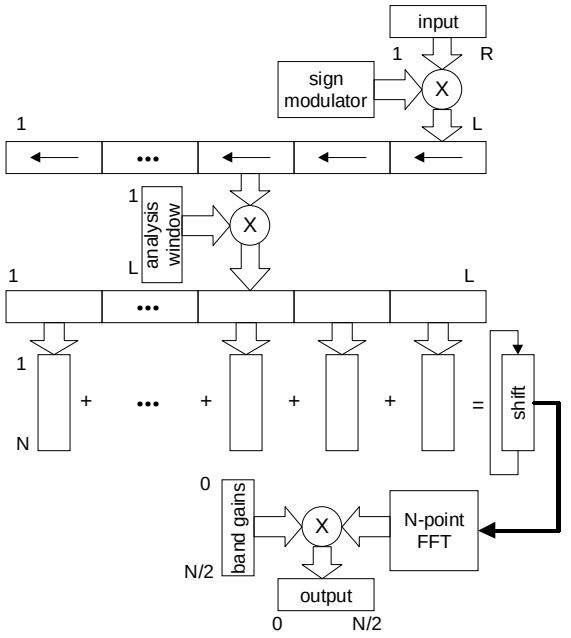


**Figure 2. WOLA analysis processing**

Analysis processing (Figure 2) shifts an *L*-sample window along the input FIFO, R samples at a time. Following this, the signal is "folded," circularly shifted and processed with an *N*-point FFT. Typical values are *R=8*, *L=128* and *N=32*, resulting in *N/2+1* complex frequency domain values computed every *R* samples.

Synthesis processing (Figure 3) is the inverse of analysis processing. The modified frequency domain data is processed with an inverse FFT, circularly shifted and periodically extended. Then, the synthesis window is applied and the overlap-add operation is performed.

The filterbank oversampling rate (*OS*) is *OS=N/R* (i.e., FFT size / Input block size). The synthesis window is also decimated by OS. Thus, oversampling reduces group delay in two ways: (i) the group delay through the (FIR) synthesis window is reduced and (ii) the blocking delay is reduced when *R* is shortened. A blocking delay occurs because the system is reading an input block, processing a block and writing an output block simultaneously.

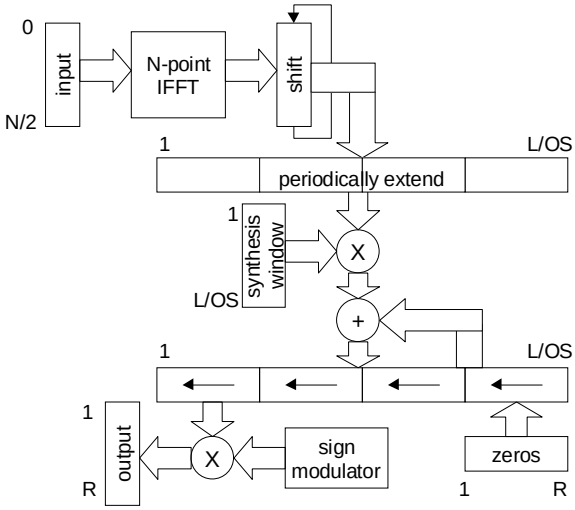


**Figure 3. WOLA synthesis processing**

The FFT size (*N*), window length (*L*) and input block step size (*R*) are all adjustable. Table 1 shows some sample filterbank configurations and the performance that can be realized. Note how reduced group delay can be "traded" for increased power consumption, reduced fidelity (a lower SFDR), or both.

**Table 1. Sample filterbank configurations (SFDR: spurious-free dynamic range; relative power for filterbank only)**

| Bands (=N/2) | OS (=N/R) | Delay (ms) | Rel. Power | SFDR (dB) |
|---|---|---|---|---|
| 16 | 2 | 14 | 1 | 65 |
| 16 | 4 | 6 | 1.5 | 50 |
| 32 | 4 | 12 | 1.6 | 45 |
| 128 | 1 | 27 | 2 | 40 |

The WOLA filterbank can operate in either even or odd stacking (Figure 4). Even stacking provides *N/2+1* bands and is useful when the highest and lowest half-bands are not processed. Odd stacking provides *N/2* bands and is useful in subband coding, where equal width bands result in simplified processing. Both even and odd stacking are useful in hearing aid applications where twice the number of band-edges allows a better fit to "sharp" hearing losses.

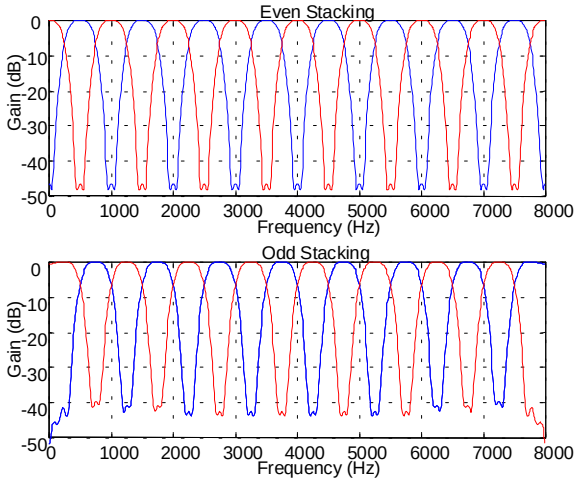


**Figure 4. Frequency responses for even and odd stacking (16-channels, $\tau$ = 6 ms, $f_s$ = 16 kHz)**

In stereo processing mode, the WOLA filterbank treats two real signals as a single complex signal. After filterbank processing, the two signals are separated by the core in a known manner [6]. Stereo mode is useful for directional processing because it provides magnitude and phase information for both input signals.

The WOLA interfaces to the DSP core via shared memory. Communication is via a control register and interrupts. The design supports concurrent processing on the WOLA and DSP core.

Within the WOLA, block floating-point arithmetic, with greater than 16-bits precision, is used to maintain fidelity. This provides more than 100 dB of numerical dynamic range. The aliasing floor is deter-

mined by the selected filterbank configuration and the window that is used (Table 1).

## 4. Input-Output Processor

The input-output processor is responsible for management of incoming and outgoing samples. Efficient management of data-flow is a key contributor to the low power consumption of our design.

Once configured, the IOP automatically performs all blocking operations on the input samples, and sends data to the DSP core and WOLA via a shared memory interface (input and output FIFOs). An interrupt is used to inform the core when the next block of data is ready for processing.

The input-output processor has integrated decimation and interpolation filters. Incoming samples are filtered by a high-order wave digital filter (WDF) before being decimated by a factor of two. The decimation filter also has an integral DC removal filter. Outgoing samples are interpolated by a factor of two and filtered by a high-order WDF before being sent to the D/A converter.

## 5. DSP Core

The DSP core is a dual-Harvard design with three computation units: a multiplier-accumulator and two address generation units. Instructions were added to support fast normalization and de-normalization. Using these instructions and ROM lookup tables, common audio processing operations like $\log_2(x)$, $2^x$, sqrt(x) and 1/x can be efficiently calculated – typically in 10 to 12 cycles.

Memory consists of two 1 kword data spaces and a 2 kword program memory space. Additional shared memory for the WOLA filterbank and IOP is also provided. The core provides 1 MIPS/MHz operation and has a maximum clock speed of 4 MHz at 1 volt. At 1.8 volts 30 MHz operation is possible.

## 6. Mixed-Signal Interface

The mixed-signal chip employs three micropower converters: two 14-bit A/D converters and one 14-bit D/A converter [5] that each support a maximum sample rate of 40 kHz (2X oversampled). The input stages have adjustable gain preamplifiers that are suitable for a wide range of signal levels. The output stage has an adjustable output attenuator. A six-input, low-speed 10-bit A/D converter is provided for sampling low-speed signals (e.g., user controls). The entire mixed-signal chip is under software control of the DSP core via a low-speed synchronous serial interface.

An on-chip oscillator provides a 1.28 MHz clock for low power audio applications. For more demanding applications, an off-chip oscillator can be used. A charge pump is provided to support the use of 1.8 volt $E^2$PROMs and for operating the system in more demanding applications at higher clock speeds.

## 7. Results

The design is implemented as a two-chip system: the digital chip is fabricated on 0.18 μm CMOS; the mixed signal chip is fabricated on 1 μm CMOS.

Prototype versions of the chipset are packaged into a 6.5 x 3.5 x 2.5 mm hybrid circuit. A 7 x 5 x 1.5 mm multi-chip module package has also been developed.

The mixed-signal chip works down to 0.9 volts; prototypes of the digital chip are operating at 1.6 volts. From a 1.2 volt cell (using the 1.8 volt charge-pump for the digital chip), we have demonstrated a complex audio processing algorithm (80% core utilization), that results in a system current consumption of only 1.8 mA. This algorithm requires 5-6 MIPS, which results in 0.25 mW/MIPS operation. The WOLA filterbank alone provides 0.18 mW/MIPS operation.

A production version of the digital chip is nearly complete. This version operates at 1 volt ± 10% and consumes less that 500μA. Reduced size and 0.125 mW/MIPS system operation will be achieved with this version.

## 8. Applications

A number of audio processing algorithms including equalization, dynamic range compression [7], directional processing and noise reduction have already been implemented. All are suitable for audio front-end processing in telecom applications, speech recognition systems or hearing aids. Stereo mode is proving particularly useful for frequency-domain directional processing.

Critically sampled sub-band CODECs can be realized via decimation in the frequency domain [4]. We have developed a CODEC that provides quality suitable for speech recording. The miniature size and low power consumption of our design allows the creation of miniature audio playback and recording devices. This CODEC can also be used in digital RF source coding applications.

The small size and low power broaden the range of applications where DSP is practical. For example, signal conditioning, data compression and data transmission could easily be embedded into a smart sensor.

## 9. Conclusions

Ultra low-power DSP systems will result in many new DSP applications such as hearing aids, personal digital assistants and portable audio playback devices. Our design demonstrates that ultra low-power DSP systems can offer sufficient computational capability, flexibility and miniaturization to be used in these applications.

In our design, the greatest power savings results from an architecture that "codes" a frequently used part of the algorithm (filtering) in an efficient hardware implementation. Additional power savings come from the use of deep sub-micron technology, low-voltage operation, adherence to a low-power design methodology and a careful system partitioning. A 16-bit DSP core combined with the WOLA filterbank provides the flexibility needed for a wide range of algorithms.

During the development of this design, we found that close collaboration between DSP algorithm designers and low-power chip designers was crucial for achieving a successful design that satisfied the needs of the end application.

## References

[1] Brennan, R.L., Schneider, T., "A Flexible Filterbank Structure for Extreme Signal Manipulations in Digital Hearing Aids," *Proc. ISCAS-98*, Monterey, CA, June 1998.

[2] Crochiere, R.E. and Rabiner, L.R., *Multirate Digital Signal Processing*. Prentice-Hall Inc., 1983.

[3] Drollinger, Andreas et al., "Efficient Subband Coder Implementation for Portable Low-power Applications," *Proc. Int'l COST 254 Workshop*, Neuchâtel, Switzerland, May 1999.

[4] Heller, P.N., Karp, R., Nguyen, T.Q., "A General Formulation of Modulated Filter Banks," *IEEE Trans. on Signal Proc.*, Vol. 47, No. 4, pp. 986-1002.

[5] Heubi, A., Balsiger, P., Pellandini, F., "Micropower Relative Precision 13-bit Cyclic RSD A/D Converter," *Proc. 1996 IEEE Int'l Symposium on Low Power Electronics and Design*, Monterey, CA, August 1996, pp. 253-257.

[6] Press et al., *Numerical Recipes in C*, Cambridge University Press.

[7] Schneider, T., Brennan R.L., "A Multichannel Compression Strategy for a Digital Hearing Aid," *Proc. ICASSP-97*, Munich, Germany, pp. 411-415

[8] Vaidyanathan, P.P., *Multirate Systems and Filter Banks*. Prentice-Hall Inc., 1993.