

Planning for Success in Real-Time Acoustic Processing

David Katz, Infotainment Systems Architecture Director

Low latency, real-time acoustic processing is a key factor in many embedded processing applications, among them voice preprocessing, speech recognition, and active noise cancellation (ANC). As real-time performance requirements steadily increase within these application domains, developers need to adopt a strategic mindset to properly accommodate these needs. Given the substantial performance offered by many larger systems on chips, it may be tempting to simply load these devices with any additional tasks that arise, but it's important to understand that latency and determinism are critical elements that can easily lead to major real-time system problems if not considered carefully. This article will explore issues designers should consider when choosing between an SoC and dedicated audio DSP in order to avoid unpleasant surprises in their real-time acoustic systems.

Low latency acoustic systems cover a wide range of applications. For instance, in the automotive space alone, low latency is critical for personal audio zones, road noise cancellation, and in-car communication systems, to name a few.

With the emerging trend of vehicle electrification, RNC becomes even more important because there's no combustion engine generating noticeable noise. Therefore, the sounds associated with the car-to-road interface become much more noticeable and offensive. Reducing this noise not only creates a more comfortable riding experience, but it also reduces driver fatigue. There are numerous challenges associated with implementing a low latency acoustic system on an SoC as opposed to on a dedicated audio DSP. These include issues of latency, scalability, upgradeability, algorithm considerations, hardware acceleration, and customer support. Let's examine each of these in turn.

Latency

In

The issue of latency in real-time acoustic processing systems is an important one. If the processor cannot keep up with the real-time data movement and computational demands of the system, unacceptable audio drops can result.

Typically, SoCs have small on-chip SRAMs and, accordingly, must rely on cache for most local memory accesses. This introduces nondeterministic availability of code and data, and it also increases processing latency. For a real-time application such as ANC, this alone can be a deal breaker. However, there's also the fact that SoCs run non-real-time operating systems that manage heavy multitasking loads. This amplifies the nondeterministic operating characteristic of the system, making it very difficult to support relatively complex acoustic processing in a multitasking environment.

Figure 1 shows a concrete example of an SoC running a real-time audio processing load, where CPU loading spikes as higher priority SoC tasks are serviced. These spikes may occur, for example, due to SoC-centric activities such as

VISIT ANALOG.COM

media rendering, browsing, or app execution on the system. Whenever the spikes surpass 100% CPU loading, the SoC is no longer operating in real time, and this will result in audio dropouts.



Figure 1. Instantaneous CPU loads for a representative SoC running high audio memory processing in addition to other tasks.¹

Audio DSPs, on the other hand, are architected for low latency throughout the signal processing path, from sampled audio input to composite (for example, audio + anti-noise) speaker output. L1 instruction and data SRAM, the single-cycle memory closest to the processor core, is ample enough to support many processing algorithms without offloading intermediate data to off-chip memory. Additionally, on-chip L2 memory (farther from the core but still much faster access than off-chip DRAM) helps provide a buffer for intermediate data operations when L1 SRAM storage is exceeded. Finally, audio DSPs commonly run a real-time operating system (RTOS) that ensures incoming data can be processed and sent out to its target destination before new input data arrives, thus ensuring that data buffers don't overflow during real-time operation.

The actual latency in system boot—often measured by time-to-audio availability can also be an important metric, especially in automotive systems where audible warnings must be broadcast within a certain window from startup. In the SoC world, where it's typical to have a lengthy boot sequence that involves bringing up the operating system for the entire device, it can be difficult or impossible to meet this start-up requirement. On the other hand, a standalone audio DSP running its own RTOS unaffected by other extraneous system priorities can be optimized for fast boot that comfortably satisfies the time-to-audio requirement.

Scalability

While latency concerns are problematic for SoCs in applications such as noise control, another key shortcoming for SoCs aspiring to perform acoustic processing is in scalability. In other words, SoCs that control large systems (such as automotive head-end units and clusters) with many disparate subsystems cannot easily scale from low end to high end audio needs because there is constant conflict between the scalability needs of each subsystem component, requiring trade-offs in the overall SoC utilization. For instance, if a head-end SoC connects to a remote tuner and, across automotive models, that tuner needs to scale from a few channels to many channels, each channel configuration will amplify the real-time concerns mentioned earlier. This is due to each additional feature under the SoC's control changing the real-time behavior of the SoC and the resource availability of key architectural components used by multiple functions. These resources include aspects such as memory bandwidth, processor core cycles, and system bus fabric arbitration slots.

Apart from the concern about other subsystems connecting into the multitasking SoC, the acoustic subsystem itself has its own scalability issues. There's a low end to high end scaling (for example, increasing the numbers of microphone and speaker channels in an ANC application), and there's also the audio experience scaling, from basic audio decode and stereo playback up through 3D virtualization and other premium features. Though these requirements do not share the real-time constraints of ANC systems, they nonetheless relate directly to the choice of audio processor for a system.

Utilizing a separate audio DSP as a coprocessor to an SoC is a perfect solution to the audio scalability problem, enabling modular system design and a costoptimized solution. The SoC can focus much less on the real-time acoustic processing needs of the larger system, instead offloading that processing to the low latency audio DSP. Moreover, audio DSPs that offer several different price/ performance/memory levels across a comprehensive code-compatible and pincompatible roadmap offer maximum flexibility for system designers to right-size the audio performance offering for a given product tier.



Single-Core DSP for Scalable High Performance Audio

Single-Core SHARC+ Performance with Scalability
from 400 MHz to 1 GHz

Scalable On-Chip SRAM (Up to 1.6 MB)

Generic	Speed Grades	On-Chip SRAM		DDR3/L	Package Type (T _{junction} 125°C)
ADSP-21569W	1 GHz	640 kB L1	1 MB L2	Yes	400-Ball BGA 17 mm × 17 mm (0.8 p)
	800 MHz				
ADSP-21567W	800 MHz		512 kB L2		
	600 MHz				
ADSP-21566W	400 MHz		256 kB L2		
ADSP-21565W	1 GHz	640 kB L1	1 MB L2	N/A	120-Lead LQFP-EP 14 mm × 14 mm
	800 MHz				
ADSP-21563W	800 MHz		512 kB L2		
	600 MHz				
ADSP-21562W	400 MHz		256 kB L2		

Figure 2. ADSP-2156x DSP, illustrative of a highly scalable audio processor.

Upgradeability

As over-the-air firmware updates become more common in today's vehicles, upgradeability to issue critical patches or provide new functionality becomes increasingly important. This can cause major issues for an SoC because of the increased dependencies among its various subsystems. First, on SoCs, multiple processing and data movement threads are vying for resources. This increases competition for processor MIPS and memory when new features are added, especially during bursts of peak activity. From the audio perspective, feature additions in other SoC control domains can have an unpredictable effect on real-time acoustic performance. One side effect of this situation is that new functionality must be cross-tested across all operating planes, resulting in myriad permutations between various operating modes of the competing subsystems. Thus, software verification increases exponentially for each upgrade package.

Viewed from a different angle, it could be said that improvements to SoC audio performance are dependent on available SoC MIPS, in addition to the feature roadmaps for the other subsystems controlled by the SoC.

Algorithm Development and Performance

It should be apparent that, when it comes to developing real-time acoustic algorithms, audio DSPs are purpose-built for the task. As a significant differentiator to SoCs, standalone audio DSPs can offer graphical development environments that allow engineers with minimal DSP coding experience to add quality acoustic processing into their designs. This type of tool can lower development costs by reducing development time without sacrificing quality or performance. As an example, ADI's SigmaStudio[®] graphical audio development environment offers a wide variety of signal processing algorithms integrated into an intuitive graphical user interface (GUI), allowing the creation of complicated audio signal flows. It also supports graphical A²B configuration for audio transport, greatly helping to catalyze real-time acoustic system development.

Audio-Friendly Hardware Features

In addition to a processor core architecture that's specifically designed for efficient parallel floating-point computations and data accesses, audio DSPs often have dedicated multichannel accelerators for common audio primitives such as fast Fourier transforms (FFTs), finite and infinite impulse response (FIR and IIR) filtering, and asynchronous sample rate conversion (ASRC). These allow real-time audio filtering, sampling, and frequency domain conversion outside of the core CPU, increasing the effective core performance. Additionally, they can facilitate a flexible and user-friendly programming model due to their optimized architecture and dataflow-management capabilities.

Because of the proliferation of audio channel counts, filter streams, sampling rates, and the like, it's important to have a maximally configurable pin interface that allows in-line sample rate conversion, precision clocking, and synchronous high speed serial ports to route data efficiently and avoid added latency or external interface logic. The digital audio interconnect (DAI) of ADI's SHARC[®] family processors exemplifies this capability, as shown in Figure 4.



Figure 3. Analog Devices' SigmaStudio graphical development environment.



Figure 4. Digital audio interconnect (DAI) block diagram.

Customer Support

One often overlooked aspect of developing with an embedded processor is customer support for the device.

Although SoC vendors promote running acoustic algorithms on their integrated DSP products, this carries several liabilities in practice. For one, vendor support is often more complex, since acoustic expertise is not typically the domain of the SoC application development. Consequently, there tends to be weak support for customers seeking to develop their own acoustic algorithms on the SoC's on-chip DSP technology. Rather, the vendor may offer standard algorithms and charge a significant NRE to port acoustic algorithms to one or more cores of the SoC. Even so, there's no guarantee of success, especially if the vendor doesn't offer mature, low latency framework software. Finally, the third-party ecosystem for SoC-based acoustic processing tends to be rather fragile, since it's not a focus of the SoC, but rather an opportunistically supported feature.

Clearly, a purpose-built audio DSP carries with it a much stronger ecosystem for development of complex acoustic systems, from optimized algorithm libraries and device drivers to real-time operating systems and easy to use development tools. What's more, audio-focused reference platforms (like ADI's SHARC audio module platform, shown in Figure 5) that speed time to market are a rarity for SoCs, but quite common in the standalone audio DSP domain.



Figure 5. SHARC audio module (SAM) development platform.

In sum, it is apparent that designing real-time acoustic systems involves deliberate, strategic planning of system resources and cannot simply be managed through allocating leftover processing headroom on a multitasking SoC. Instead, a standalone audio DSP optimized for low latency processing is likely to lead to increased robustness, decreased development time, and optimal scalability to accommodate future system needs and performance tiers.

References

¹Paul Beckmann. "Multicore SOC Processors: Performance, Analysis, and Optimization." 2017 AES International Conference on Automotive Audio, August 2017.

About the Author

David Katz has 30 years of experience in analog, digital, and embedded systems design. He is director of systems architecture for automotive infotainment at Analog Devices, Inc. He has published internationally close to 100 embedded processing articles, and he has presented several conference papers in the field. Previously, he worked at Motorola, Inc., as a senior design engineer in cable modem and factory automation groups. David holds both a B.S. and an M.Eng. in electrical engineering from Cornell University. He can be reached at david.katz@analog.com.

Engage with the ADI technology experts in our online support community. Ask your tough design questions, browse FAQs, or join a conversation.



Visit ez.analog.com



For regional headquarters, sales, and distributors or to contact customer service and technical support, visit analog.com/contact.

Ask our ADI technology experts tough questions, browse FAOs, or join a conversation at the EngineerZone Online Support Community. Visit ez.analog.com.

©2021 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners. VISIT ANALOG.COM

TA22758-4/21