cadence®

How to Shorten and Simplify Embedded Audio Product Creation

By Paul Beckmann, Ph.D., DSP Concepts, and Gerard Andrews, Cadence

From concept through research and development to tuning, creating audio products is tough work. Product complexity is growing and so, too, are consumer expectations. Yet, the product development process hasn't changed much over the years, which results in inefficiencies in terms of effort and cost, ultimately reducing time to market. In this paper, we will discuss what you need to know in order to efficiently develop the embedded audio system for today's key application areas, including automotive and consumer.

Contents

Introduction 1

Where Are All of the DSP Engineers?.................................1

Product Concept 3

Product Research..............................3

Product Development.......................3

Product Tuning.................................4

What Is Needed to Improve Product Development?.....................4

Summary..4

Introduction

Sound is a part of more application areas than ever before. Consider today's typical car—there's the music and entertainment system (CD player, traditional and/or satellite radio, MP3 player, DVD player), the navigation system, BlueTooth connectivity with the smartphone, and typical alert-related sounds that chime when, for instance, the headlights are left on or the trunk door is ajar. If it's an electric car, the vehicle might synthesize engine noises to alert nearby pedestrians. In fact, a car now typically boasts 10 to 15 different audio algorithms controlling each of these different sound-related functions. Moreover, sounds are not simply released in a vehicle; voice command and control functions, such as hands-free calling, mean that always-on, always-listening capabilities are needed to act on user-generated sounds.

The four main stages of product development (concept, research, development, tuning) must account for how each of the audio functions will work on their own and, in many cases, together. As we've mentioned, the overall process for moving through each of these stages has been stuck in the proverbial Stone Age. OEMs are typically so focused on getting the next product out the door that there's little time to investigate tools that could streamline the process.

Before we go into more detail about the product development stages, let's take a closer look at the technical roles behind developing a product featuring audio. There are clearly some challenges that are driven by knowledge/expertise gaps in the engineering community.

Where Are All of the DSP Engineers?

Working through a product development effort are the DSP engineer, the audio engineer, and the embedded software engineer. Now, there are plenty of embedded software engineers around, and it's not difficult to find talented audio engineers, either. However, qualified audio DSP engineers—the ones who actually write the code to perform a specific function, such as noise cancellation or beamforming—are more difficult to find. In addition, the intersection of these three disciplines is even more rare. (See Figure 1).

Audio DSP engineers must understand the math and algorithms behind each audio function. They need expertise with audio processing blocks such as EQ, echo cancellation, and noise suppression. Familiarity with the audio characteristics of speakers and microphones is required. An understanding of acoustics design doesn't hurt, either!

Compounding this problem is the sub-specialty of audio DSP, which takes into account not only the mathematical plotting of the different signal-to-noise ratio (SNR) and signal manipulations, but the understanding of the human psychoacoustic responses on how to handle those signal manipulations. A textbook understanding of DSPs alone is not good enough to deliver an optimal audio system.

It is also important to note that initial software implementations are typically written in generic C code. In order to achieve the desired performance, this code must be further optimized and, in older DSP architectures, even ported to assembly language. But there are fewer engineers studying the mathematical foundations of algorithms required to complete these optimizations. At the college level, students tend to be more interested in studying higher level languages, so they graduate with little experience in low-level software.

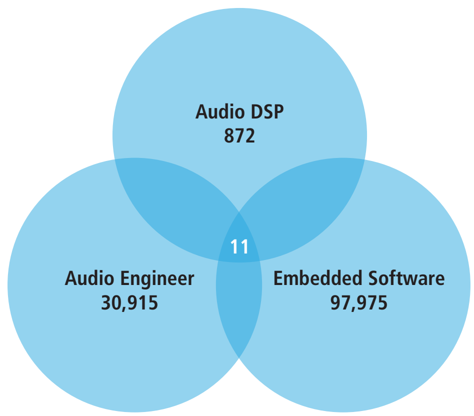

Figure 1: The complexity of audio systems is surpassing the available skill sets to create these systems. As this chart indicates, based on a search of LinkedIn profiles, there's a relatively small number of audio engineers who are well versed in DSP audio theory and embedded software design. Source: DSP Concepts

How are OEMs managing the knowledge gap? Often, DSP engineers end up overloaded and, as a consequence, are forced to make suboptimal design decisions. The audio engineers—who have the aural skills to evaluate audio quality—are often not involved in the process by the DSP engineers and, as a result, can't get exactly what they need for optimal sound. Designs are deemed to be "good enough" and get shipped in this condition (which explains why there are so many MP3 player docking stations on the market with subpar sound!).

Now, if subpar quality is OK with you, then the dearth of talented DSP engineers might not be worrisome. But if you want to go beyond "good enough" in terms of quality with a development process that is smooth, reliable, and efficient, then you'll want to fix what is now a broken process that relies too much on a skill set that is lacking.

In the next sections of this paper, let's take a closer look at each of the product development stages (Figure 2).

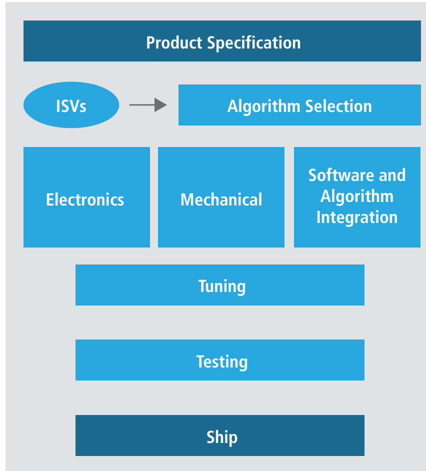

Figure 2: Traditional product development process

Product Concept

At the earliest stage of product development, product marketing and product engineering come together to discuss and decide on top-level hardware- and software-based features for a product. Will the new MP3 player need wireless headphones and wireless speakers? Should it interface with a smartphone? Should users be allowed to control the bass and treble? The list of considerations goes on for each product, as the team evaluates and refines a feature list based on what prospective users want as well as what potential users don't yet know they need.

Cost, time to market, and available resources to implement the product concept are key considerations at this stage.

Product Research

Concept development naturally leads to the research stage, where the DSP engineer gets more involved. Here, the DSP engineer evaluates the hardware options and constraints in terms of criteria such as features, memory, cost, and power consumption. The engineer also determines what kind of software is needed to enable the desired functions, and whether the software exists or needs to be written. Third-party algorithms are an option here, as is in-house algorithm development. Time, effort, and cost are deciding factors.

Now, as we've noted, audio DSP engineers have a lot of responsibilities on their plates. Yet, performing a thorough IP evaluation is time consuming. For example, say you're considering noise-cancellation IP from a few third-party vendors. In addition to determining which IP will produce the best audio quality for your design, you'll need to answer, for each IP option, an array of questions that affect the overall design: What is the CPU load for the algorithms? What is the memory requirement? What clock speed is needed? This evaluation and decision process continues for each audio algorithm required to provide the desired functionality.

Typically, by the time this stage is underway, the processor for the design has already been chosen. Hardware and software development are happening in parallel, which impacts the end product. So, as you evaluate your IP options, you might find that the processor isn't fast enough, so you must give up features or sound quality. For instance, high-quality "upmixing," converting from a two-channel stereo file to a multi-channel surround output, requires careful manipulation of the audio signals without losing any information. To put that in perspective, by comparison, a dropped frame in a video application is usually unnoticeable to viewers. However, when it comes to audio, the ears are sensitive enough to notice the pops or clicks that are a result of missing data. Similar decisions and tradeoffs must be made for each IP that you evaluate or create in house.

Product Development

At this stage, the embedded systems engineer has the chip integrated onto the board and conducts testing and validation exercises to ensure that everything works well together. This engineer's work continues as the board is integrated into the end product. Through this effort, the choices that the embedded engineer makes determine (and limit!) which features will or won't make it into the final product. With traditional development methods, it's only after the process of fitting everything in that the engineer can determine if there is any processing power "left over" to enable the product to be further differentiated with additional features.

Product Tuning

This stage is where the audio engineer—the one with the "golden ear"—ensures sound quality of the audio device. Doing so involves tweaking and tuning multiple parameters in real time. Now, different audio functions have different sample rates, and various components in the end product can impact sound quality. Consider our automotive example—you can't really tune an automotive audio device until the car is fully manufactured. Even carpeting—before and after it has been installed—can change the way that the sound sounds.

Every product category requires some level of last-minute tuning. How can you guard against tuning and then breaking some aspect of the design—especially this late in the development cycle?

Another consideration is that sound-quality comparisons are optimal when they are performed almost instantaneously. Often, however, DSP engineers must rebuild and compile code for different sound settings. So by the time an audio developer listens to a second or a third version, he or she likely won't have the same fresh recollection of how the first one sounded.

What Is Needed to Improve Product Development?

As you can see, each stage of the product development process has challenges in terms of skill sets, time, and effort required. The skill sets required are unlikely to change any time soon. As for time, development schedules are probably only going to get tighter, especially when designing for the ultra-competitive consumer market. Effort, however, can be enhanced, given the right tools.

What is needed to streamline and accelerate the entire audio product development process is a system that allows you to quickly design your audio system, prototype it on a board, test, and then deploy. Each engineer involved in the project must be able to collaborate, develop in parallel, and tune the design in an efficient matter. They should have a fast way to evaluate and compare the performance of different audio algorithms on a target processor. Ideally, the design team should also be able to focus on design optimization and differentiation versus low-level coding.

An effective solution would provide a graphical design environment that would bring together the different engineering functions: audio engineering, DSP engineering, and embedded software engineering. A graphical environment would allow individual functions to be represented by "blocks" of code that can be arranged and re-arranged by anyone on the team. The graphical tool should also have real-time tuning capabilities so that code does not need to be changed and re-compiled before it can be heard again. Such real-time tuning capabilities would enable the audio engineer to optimize the sound of the DSP design. In addition, an ideal system would provide a large library of audio functions, pre-optimized for efficiency. This way, design teams would not be hampered by any areas of insufficient expertise. An ideal solution should also allow the design team to quickly evaluate processing resources, audio algorithms, and IP in order to streamline the development cycle and, where possible, reduce development risks and costs.

Summary

While integral to many application areas, the specifications for audio signal processing vary widely, from automotive applications with multiple audio sources to always-on, always-listening electronic devices that require low energy consumption. An effective audio design and prototyping solution can go a long way in taking the product development process out of the Stone Age.