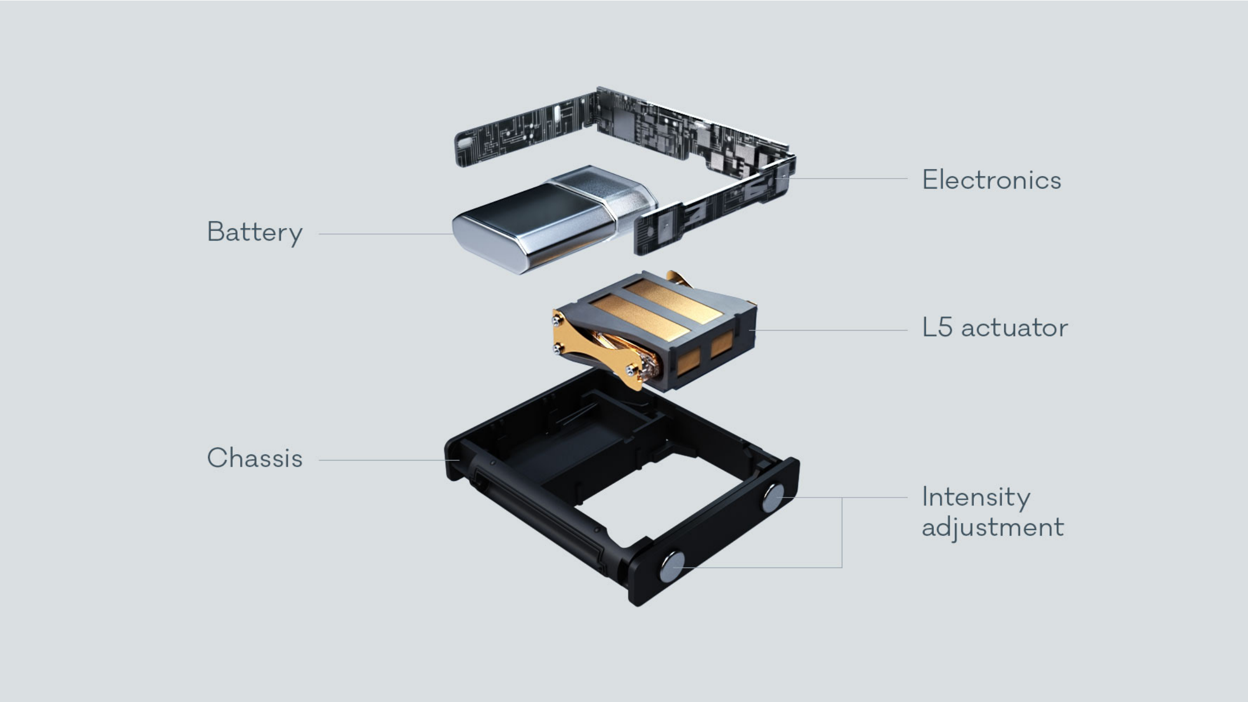
Best Practices for Implementing Realistic Vibrotactile Haptics

Streamlining Integration of Sophisticated Haptics into Your Hardware



Incorporating realistic vibrotactile haptics into your hardware platform can present unique design and engineering challenges. Compared with basic haptic systems that use only simple motors, sophisticated haptic solutions are systems that can include wide-band actuators and digital signal processing (DSP). These systems provide an array of new design options but also introduce new levels of complexity. To help ensure successful, efficient integration, you need to consider an expanded set of factors for product design, hardware implementation, software implementation, and testing and calibration.

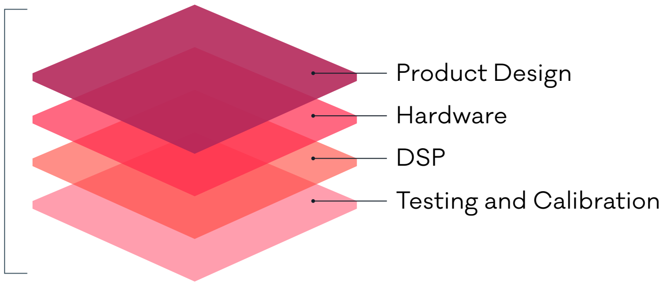
While the quality level required by the haptic system components has been defined in a specification proposal VT-1 (available as a white paper on our website), there is only limited practical guidance available for integrating realistic vibrotactile haptics into hardware platforms.

For example, Apple has published a concise set of human interface guidelines, information about its Core Haptics engine and a few other articles on haptics to help developers take advantage of the haptic possibilities available with iPhones. But, since Apple produces its own devices, it does not share recommendations for designing haptics-enabled hardware.

What if you plan to integrate vibrotactile haptics into headphones, game controllers, wearables or other types of devices? As a product designer, where should you begin?

Start planning the integration with a few best practices designed for the latest haptic systems. With the right steps, drawn from extensive experience, you can simplify product integration and reduce risk in production.

Your Guide to Realistic Quality Haptic Integration



Organize planning for realistic haptic integration around four key areas: product design, hardware implementation, software implementation, and testing and calibration. Each area brings a new set of challenges, considerations and best practices compared to basic haptic integration.[1]

1. **Product design:** Envision the user experience and examine your options before you begin prototyping.

2. **Hardware implementation:** Identify potential challenges with placement of haptic actuators, electromagnetic interference, wireless optimization and more.

3. **Software-based DSP implementation:** Determine whether you should incorporate a discrete processor for running DSP and understand why you should tap into existing digitized audio to generate haptic signals.

4. **Testing and calibration:** Establish processes for testing actuators, updating firmware, recalibrating software and conducting any last fine-tuning before your product hits the shelves.

This guide offers recommendations and best practices for each of these areas.

[1] For clarification, we refer to more simple, single-frequency vibrotactile feedback that does not meet the VT-1 specification as "basic haptics" in this document.

1. Designing the Product

As you begin to design your next game controller or a new model of headphones, keep in mind that integrating a sophisticated haptic system into your device requires more preparation than attaching the type of simple motors used for basic haptics. By presenting more options than basic haptic systems, realistic systems require you to make more design decisions. Taking the time to envision the UX and fully evaluate your implementation options before you start prototyping can help you maximize the impact of realistic haptics in your device and enhance the efficiency of the entire implementation process.

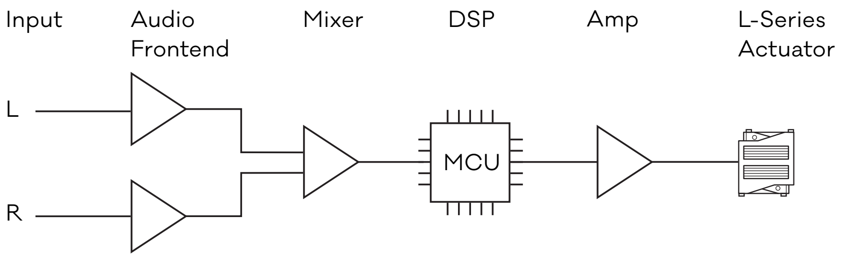
1.1. Envision the UX

What do you want your device to do? How should users interact with the haptic capabilities? Following a few key recommendations at this stage can help you streamline product design and development.

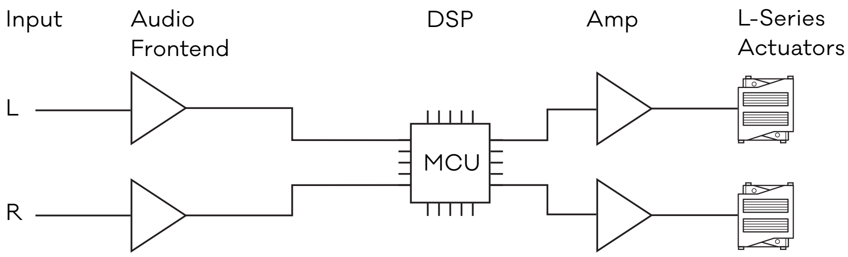
Audio source
Whether you are designing a pair of headphones or a game controller, a stereo haptic system – which uses stereo audio to drive two actuators – can create a more nuanced, immersive experience than a mono system. Implementing a stereo system, however, requires a more complex design. For example, you'll need to use two independent amplifiers – one for each actuator – to produce distinct vibrations on left and right sides.

You'll also need to decide which audio elements to include in your haptic signal. In the gaming use-case, you might want to omit the communication channel, for example, and focus haptics on the soundtrack and special effects.

Mono System



Stereo System



A stereo haptic system uses stereo audio to drive two actuators, creating a more nuanced, immersive experience than a mono system.

Flexibility
Are you creating a device for a single use case or multiple use cases? You might want to design a new model of headphones that will be appealing not only for gaming but also for listening to music and watching videos. If you design for multiple use cases, you should provide a simple way to adjust the experience for each distinct application.

User control
Enable users to control their haptic experience. You might allow them to turn off haptics or adjust the intensity so they can rebalance audio and haptics according to their own preferences. If you provide stepped haptic control, you do not need to include as many steps as you would for audio – human tactile perception is less sensitive to gradations than auditory perception.

You could also provide user-selectable haptic profiles for different types of content, such as games, videos and music. Users might prefer an intense haptic experience for games, a dynamic experience for videos and a more subtle experience for music.

To simplify user operation, provide physical controls and mount them on the outside of the device. Requiring the user to adjust settings on a computer or through a software interface could be inconvenient and frustrating.

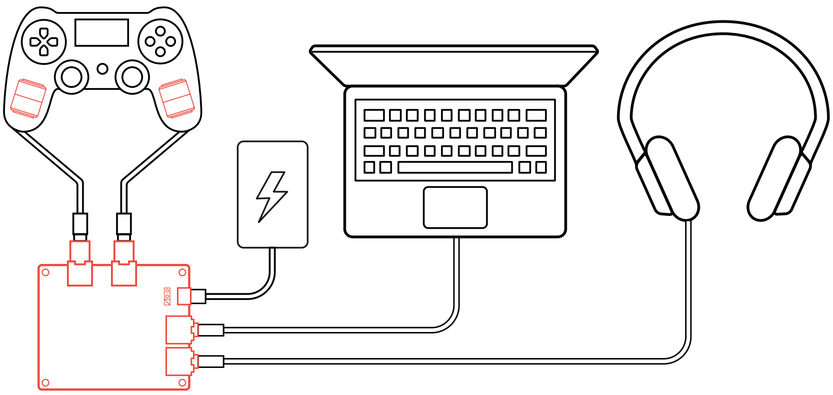
Differentiation
Selecting a VT-1 compliant haptic system with wideband actuators and DSP can help you provide a more immersive experience than more basic haptic systems. But with a software-controlled system, you also have an important opportunity to create a signature haptic character to differentiate your brand. Just as various headphone manufacturers might be known for a bass-rich or punchy sound, you can establish a unique haptic feel for your products.

1.2. Start prototyping

With basic haptic systems, you can generally anticipate the kind of notification-like feedback your device will deliver even before you create a product prototype. Limited implementation options and relatively simple hardware components do not leave much room for experimentation or optimization.

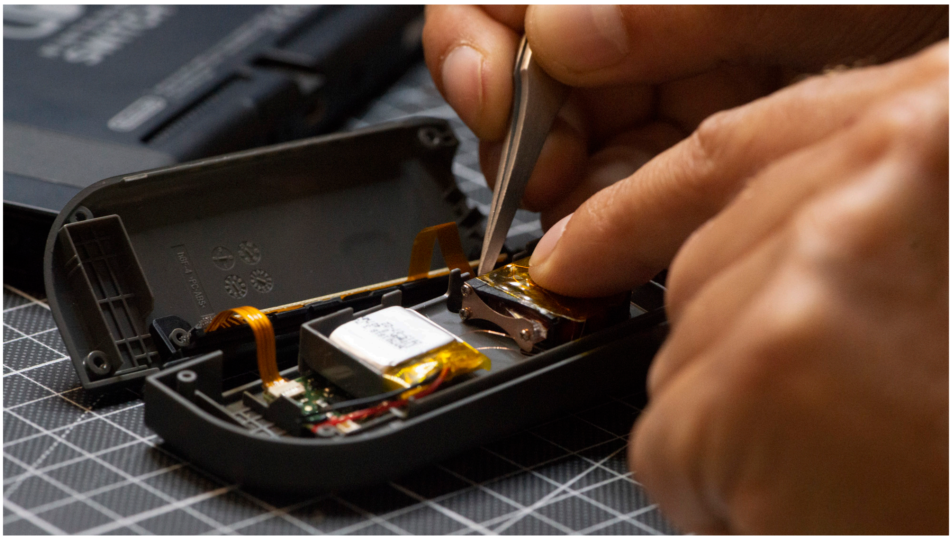
When you decide to implement realistic haptics, however, prototyping plays a more critical role in product design. Prototyping enables you to explore the wide variety of user experiences you can create, assess all of your implementation options and identify any potential issues with this more complex haptics system. Creating prototypes before constructing schematic designs can help spark new ideas and enable you to avoid surprises later.

To begin prototyping, use the evaluation kit offered by your prospective haptic technology partner. The kit should give you everything you need, including both hardware and software.



The Lofelt Wave evaluation kit includes DSP firmware plus haptic actuators, enabling you to explore integrating sophisticated haptics into your product.

2. Developing the Mechanical and Hardware Systems



Choosing an audio-driven realistic haptic system that incorporates wideband hardware actuators can help you create a device that delivers much more realistic user experiences than basic haptic systems. Selecting this type of haptic system can also have important implications for hardware integration. In particular, optimizing actuator placement, minimizing electromagnetic interference, ensuring stereo actuators are in phase and addressing wireless system challenges will be critical to avoiding problems and delivering the best user experience.

2.1. Placement and mounting

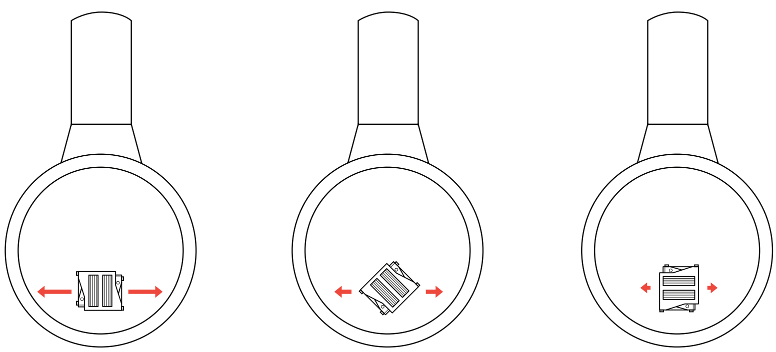
Proper placement and mounting of wideband actuators within your hardware device housing are essential for maximizing the impact of realistic feedback. For example, if you are designing headphones, you should mount the actuators in the headphone cups, close to the rigid housing, which will help transfer vibrations.

Placement is particularly important when you need to vibrate a device with a large mass. In general, actuators vibrate lighter devices more efficiently than heavier ones. But what if you want to provide tactile feedback for the trackpad on a computer? Because an actuator would be unable to adequately vibrate the entire computer, you'd need to make sure that the actuator is placed so that it focuses vibrations on the trackpad.

Correct placement and mounting are also key for minimizing unwanted vibrations within the device. Unlike the single-frequency motors used with basic haptic systems, wideband actuators vibrate across a broad frequency range. As a result, there is greater risk that one or more vibration frequencies will trigger sympathetic resonances with the device housing, internal components or mechanical parts, creating a rattling noise or a tone that is audible to users.

Modifying the design or attachment method is often the best way to prevent those sympathetic resonances. You can search for problematic frequencies during testing by employing slow frequency sweeps and by playing use-case audio (such as game audio). You can then eliminate unintended vibrations by ensuring the actuators are not touching or near to components that might vibrate.

DSP can help reduce unwanted vibrations. You can use DSP to attenuate troublesome frequencies without having to modify your hardware design. Just be aware that DSP correction might remove frequencies that are unwanted structurally but still beneficial for your particular use case. Employing DSP allows you to experiment to find the right balance between unwanted vibrations and desirable frequencies without excessive hardware redesign.



Different orientations of the actuator can significantly affect the user experience. For headphones, a horizontal orientation is optimal.

2.2. Electromagnetic interference

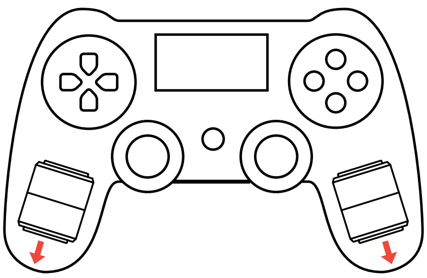
Wideband actuators that use a voice coil design, as speakers do, create strong electromagnetic fields. Consequently, the actuators must be positioned at a sufficient distance from any other components and signal routing. For products with audio, keep the haptic actuators away from audio speaker and audio signal lines. If your product has wireless capability, for example a VR controller, position the actuators away from the antennas.

The amplified signal that drives the actuator could cause this kind of electromagnetic interference (EMI). As you design your device, avoid a long cable run for that amplified signal, since a long cable run increases the likelihood of interference. If you are designing headphones, placing all the electronics on the right side and running a long cable over the headband to drive the actuator on the left side could cause interference. Instead, implement a discrete amplifier on each side and transmit only a low-power, unamplified signal over the headband. This approach could add complexity to the design, but it will help reduce the possibility of interference.

Electrical components can also produce unwanted noise that affects the haptic experience. For example, a power source might create a 50 or 60 Hz hum that affects the wideband actuator. If there is noise created by a power source, processor or switch, that noise can cause the actuator to vibrate unintentionally. This “parasitic oscillation” will create tactile feedback for the user when there shouldn’t be any. Using shielded cables, reducing the length of cables, and filtering out frequencies with DSP all can help reduce or eliminate these effects.

2.3. Phase

Because wideband actuators operate like speakers, you’ll need to ensure they are installed in phase. Each actuator moves in two directions. If you wire one actuator out of phase with the other, the quality of the haptic effects will be diminished. Polarity markings on actuators can help you make sure you are implementing them in phase.



Be sure to install actuators in phase to avoid phase cancellation of the haptic effects.

2.4. Wireless system

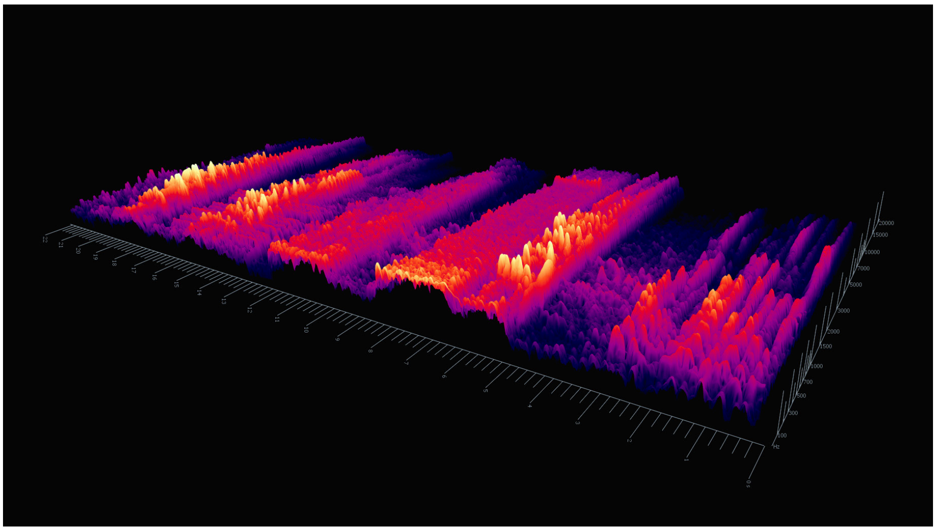
If you are producing a wireless device such as a wireless game controller, make sure audio information can be sent to and received by the device. With more basic haptic systems that produce simple rumbles, the device might receive only control information, such as motor-on and motor-off messages. A more sophisticated audio-driven haptic system relies on audio information to create the haptic experience within the device. Whether you are retrofitting an existing design for haptics or starting from scratch, be sure to incorporate audio into your device so you can provide the source for generating haptics.

With a wireless device, battery power must be another top consideration. Depending on use-case, even very efficient actuators can consume a fair amount of battery power. To provide a consistent experience to users for a reasonable amount of time, you'll have to ensure your device can supply sufficient power for the haptic system as well as other device components.

In addition, you must keep wireless latency low between the gaming console, computer or audio system and the wireless device. If there is significant wireless latency, there could be a disconnect between the audio/haptic experience and the visual experience. In a game or film, users would see an event before hearing and feeling it. As long as your product keeps wireless latency - between video and haptics - below 20 ms, you can avoid noticeable lag. If you are upgrading your wireless system to accommodate audio information – instead of sending only control messages – be sure to keep latency below that 20 ms threshold.

Finally, you'll need to minimize wireless packet loss. With audio-driven haptic systems, any packet loss that affects the audio will also affect the haptic experience.

3. Integrating DSP



Software-based DSP is a critical component of haptic systems that use audio to generate haptic feedback. Basic haptic systems send only on and off messages to their single-frequency motors. By contrast, realistic haptic systems must translate complex audio information into wideband haptic signals. The DSP engine is responsible not only for translating this audio information but also for optimizing the signal for the particular actuators in place, the type of content (such as gaming, videos or music) and the human mechanoreceptors that sense vibrations.

As a product designer, you gain tremendous flexibility by using DSP. You can use DSP to customize the UX and to quickly modify the way your device operates without extensive hardware design changes.

If you plan to implement realistic vibrotactile haptics, you and your engineering team will need to decide what hardware resources you'll use to run the DSP software and how the software will acquire the audio it processes.

3.1. Choosing a separate or existing processor for DSP

A DSP engine runs as firmware on a microprocessor. As you plan integration of a software-controlled haptic system into your hardware design, you need to include a processor with sufficient resources for DSP.

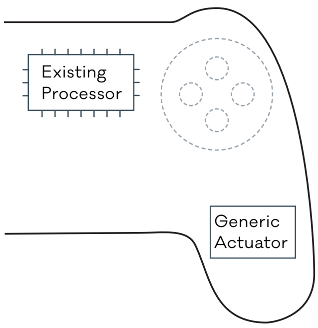
You could use a processor that is already part of your design. As long as the processor offers sufficient performance and memory – and supports the type of software library employed by the haptic system's DSP – it can be used for both DSP and other processes. Consolidating your hardware design by using a single processor can help minimize the component footprint and reduce component costs.

On the other hand, running the DSP engine on a separate processor could help accelerate implementation and minimize risks. With a separate processor, you can add this new haptic technology without significantly changing your existing design.

Importantly, if you use a separate processor, you might not have to employ a high-performance chip. Depending on the haptic system you select, you should be able to run the DSP engine on an off-the-shelf, ultra-low-power embedded Arm®-based processor. You can add this processor to your bill of materials without substantially increasing costs.

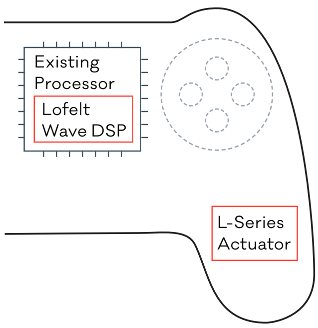
If you choose a distinct DSP processor, be sure there is a data connection between the two processors. This connection is required to support aftermarket firmware upgrades. Upgradability offers the enormous benefit of allowing you to further optimize the haptic experience and add new features even after the hardware design is set.

Basic Haptics

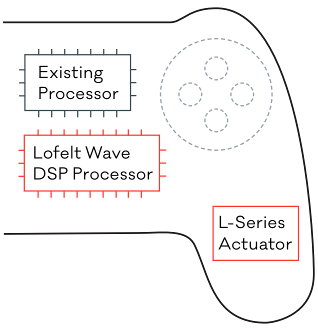


Realistic Haptics

1. Existing Processor



2. Standalone Processor

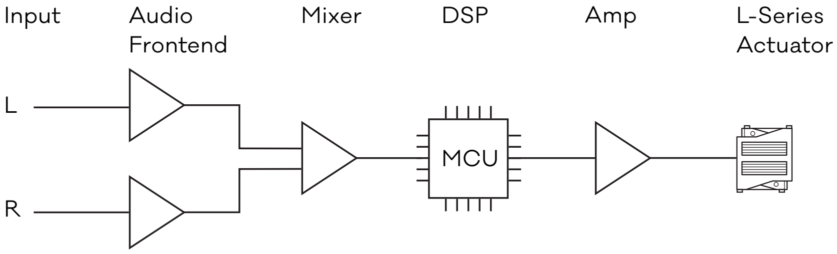


Audio-driven haptic systems require a processor for DSP. You may be able to use an existing processor, or you can employ a discrete processor.

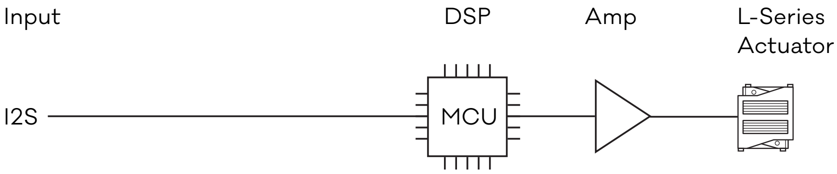
3.2. Tapping into digital audio

An audio-driven haptic system should be able to convert analog audio into a digital signal for processing. But whenever possible, design your device to tap into an existing digital (I2S) signal as the source for generating haptic experiences.

Analog System



Digital System



Tapping into digital audio can help reduce costs, protect actuators, minimize latency and more.

Using existing digital audio offers several key advantages:

Lower cost: By using existing digital information, you can avoid the costs of adding analog-to-digital (A/D) converters and other components required for preparing the audio signal.

Actuator protection: With digital information, you can easily define a maximum signal level within the software to avoid overloading – and damaging – the actuator if the audio is too loud.

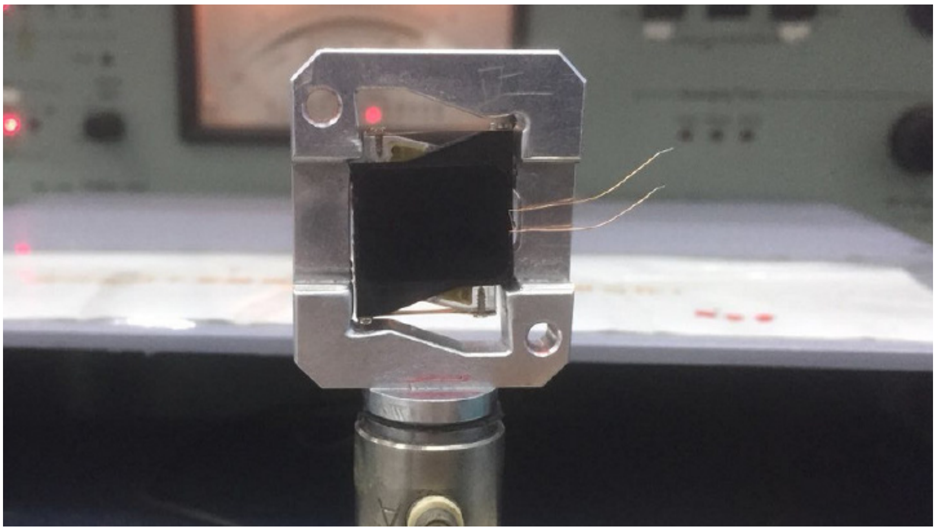
Channel independence: Using an existing digital signal makes it simpler to separate audio channels. With channel independence, you can more easily tap into soundtrack audio to create the haptic signal and avoid using the gaming communication channel, which could create unwanted vibrations.

Volume independence: Tapping into an existing digital signal also allows you to easily – and dynamically – adjust the relationship between the audio and haptic levels. You can enable users to reduce the haptic intensity for certain types of content.

Lower latency: By avoiding the conversion of analog audio into a digital signal, you can avoid adding any latency.

You can acquire digital information from a signal transmitted wirelessly. If you are producing wireless headphones, for example, the headphones receive a digital wireless signal and then convert that signal to analog to drive the speakers. Using the DSP engine, you can tap into the digital signal for processing audio without additional A/D conversion. Similarly, the DSP engine can tap into a digital signal transmitted by USB if you have a USB-wired device.

4. Testing and Calibrating



The testing and calibration processes for next-generation haptics are more complicated than the processes required for more basic haptic systems. With basic systems that use simple rotating motors, testing is often limited to confirming that the motor is rotating. With sophisticated systems that use wideband actuators, you need to ensure that the actuators are performing as they should at every frequency. You may need to alter your existing hardware testing workflow and add new processes, such as updating DSP firmware and recalibrating software, before conducting any final fine-tuning.

4.1. Testing actuators

Testing the frequency response of wideband actuators requires an accelerometer, which evaluates force in relation to frequency. Establish a process for testing the actuators during assembly, just as you would test a speaker. You can ensure that the haptic system is working as it should before finishing unit assembly.

4.2. Updating the firmware

If you are designing a new device, be sure to incorporate a means of updating the firmware during the testing process and after. Implementing a wireless update process or including an easily accessible USB port as part of your product design will enable you to update the firmware without pulling apart the unit. Updating firmware gives you the flexibility to fix bugs and add new capabilities even after the device is in the marketplace.

4.3. Recalibrating the software

The software may need to be recalibrated to match the characteristics of the actuators once they are integrated into your hardware device. You might find, for example, that the two actuators you are using in your headphones have slightly different resonant peaks. Though both might be within the manufacturing specification, you should still make an adjustment to avoid producing an unbalanced experience for users.

Or you might discover that the resonant peak of an actuator has shifted after coupling. The resonant frequency of an actuator could shift from 65 Hz to 54 Hz, for example, when coupled with a 50- or 100-gram mass. You need to ensure the actuator responds to incoming signals in the way you intended.

To compensate for these issues, first identify resonant peaks using the accelerometer. Then use DSP to optimize the signals for the actuators. By adding a filter that attenuates resonant peaks, you can deliver an even, consistent experience for users across all frequencies. Attenuating peaks also helps protect the actuators, which could otherwise become damaged if they are driven too strongly at their resonant frequencies.

4.4. Fine-tuning

You might need to make some final tweaks during engineering validation and testing (EVT) or design validation and testing (DVT) processes. Changes in product materials or the distribution of mass may all affect the haptic response. With basic haptic systems, these tweaks might require expensive, time-consuming hardware changes. With the right software-controlled haptic system, however, you should be able to make necessary adjustments using DSP. You can easily address any final issues before it's time to begin mass production.

Partnering with Lofelt

Lofelt Wave™ technology combines wideband hardware actuators and software-based DSP in an audio-driven haptic system that can deliver very natural tactile feedback. Choosing Lofelt Wave enables you to move beyond the limitations of simple, hardware-only systems and to create engaging experiences that enhance product differentiation.

Work with the Lofelt team to anticipate any design challenges and streamline integration of haptics. Lofelt team members have deep expertise across a wide range of fields, including UX design, audio electronics design, mechanical and electrical engineering, material science, multiphysics, audio technology, DSP development, prototyping and manufacturing.

Lofelt also has experience working through the complete integration process with hardware manufacturers. Lofelt Wave has been successfully integrated into products that are in the marketplace today and are delivering immersive haptic experiences for users.

Moving forward with Lofelt Wave

Need additional guidance on integrating realistic haptics into your hardware device? Contact us at lofelt.com/contact to receive Lofelt Wave spec sheets, schedule an in-person demo and discuss ways we can work together. You can also request an evaluation kit when you are ready to begin your own prototyping.