

UC San Diego

UC San Diego Previously Published Works

Title

A Wearable, Extensible, Open-Source Platform for Hearing Healthcare Research

Permalink

<https://escholarship.org/uc/item/9k50r990>

Authors

Pisha, Louis
Warchall, Julian
Zubatiy, Tamara
[et al.](#)

Publication Date

2019

DOI

10.1109/access.2019.2951145

Peer reviewed



HHS Public Access

Author manuscript

IEEE Access. Author manuscript; available in PMC 2020 June 16.

Published in final edited form as:

IEEE Access. 2019 ; 7: 162083–162101. doi:10.1109/access.2019.2951145.

A Wearable, Extensible, Open-Source Platform for Hearing Healthcare Research

Louis Pisha [Student Member, IEEE],

Department of Electrical and Computer Engineering, University of California, San Diego

Julian Warchall [Student Member, IEEE],

Department of Electrical and Computer Engineering, University of California, San Diego

Tamara Zubatiy,

Qualcomm Institute, University of California, San Diego

Sean Hamilton,

Department of Computer Science and Engineering, University of California, San Diego

Ching-Hua Lee [Student Member, IEEE],

Department of Electrical and Computer Engineering, University of California, San Diego

Ganz Chockalingam,

Qualcomm Institute, University of California, San Diego

Patrick P. Mercier [Senior Member, IEEE],

Department of Electrical and Computer Engineering, University of California, San Diego

Rajesh Gupta [Fellow, IEEE],

Department of Computer Science and Engineering, University of California, San Diego. Halicio lu
Data Science Institute

Bhaskar D. Rao [Fellow, IEEE],

Department of Electrical and Computer Engineering, University of California, San Diego

Harinath Garudadri [Member, IEEE]

Qualcomm Institute, University of California, San Diego

Abstract

Hearing loss is one of the most common conditions affecting older adults worldwide. Frequent complaints from the users of modern hearing aids include poor speech intelligibility in noisy environments and high cost, among other issues. However, the signal processing and audiological research needed to address these problems has long been hampered by proprietary development systems, underpowered embedded processors, and the difficulty of performing tests in real-world acoustical environments. To facilitate existing research in hearing healthcare and enable new investigations beyond what is currently possible, we have developed a modern, open-source hearing research platform, Open Speech Platform (OSP). This paper presents the system design of

the complete OSP wearable platform, from hardware through firmware and software to user applications. The platform provides a complete suite of basic and advanced hearing aid features which can be adapted by researchers. It serves web apps directly from a hotspot on the wearable hardware, enabling users and researchers to control the system in real time. In addition, it can simultaneously acquire high-quality electroencephalography (EEG) or other electrophysiological signals closely synchronized to the audio. All of these features are provided in a wearable form factor with enough battery life for hours of operation in the field.

I. INTRODUCTION

HEARING is essential for communication, navigation, and quality of life. The healthy ear is able to operate in a wide variety of environments over a huge dynamic range due to its highly complex nonlinear, time-varying, and attention-controlled characteristics. As a result, when hearing impairments occur, they can rarely be corrected by simply amplifying the input sound. Hearing aids (HAs) have been under development from this starting point for the last forty years, and now incorporate multi-band processing, dynamic range compression, feedback and noise management, and other advanced features.

Unfortunately, there is substantial dissatisfaction with many aspects of HAs among the user community [1]. Key factors underlying this dissatisfaction include the following:

1. *Clinical challenges:* One example is that the current best practices in HL diagnosis and intervention rely mostly on pure tone audiometry (PTA) [2], which characterize only the spectral aspects of HL, in clean conditions; the temporal dynamics in human perception of speech and music in clean and noisy environments are largely ignored. A different type of challenge is the typical need for users to see an audiologist to have fitting parameters adjusted; as an alternative, many researchers are investigating self-fitting procedures, environment-dependent profiles, and other ways to give the user control over their experience.
2. *Technical constraints:* HAs must provide sufficient battery power for processing and communication, in an acceptably small form factor, while introducing no more than 10 ms of latency [3] [4]. The overall latency requirement presents a significant challenge for noise mitigation algorithms and other advanced functions such as frequency lowering. Furthermore, binaural processing in HAs to take advantage of spatial information in noisy environments is a major challenge, because of the power requirements for wireless communication of full-band audio signals between the HAs and additional processing for adaptive beamforming.
3. *Research accessibility:* There are five major HA manufacturers: Phonak, Oticon, ReSound, Starkey, and WS Audiology. All of these manufacturers provide audiologists with tools for HA fitting, which can be used for certain kinds of clinical research. The manufacturers also sometimes provide their internal platforms for academic research in specific topics, such as directional microphones, noise management, programs for multiple listening environments,

etc. However, each of these platforms is proprietary and unique, meaning that it is difficult to generalize research across the platforms, and infeasible to modify or experiment with the algorithms in ways not intended by the manufacturers.

4. *Cost:* There is an average 8.9-year delay between HA candidacy and HA adoption, with the biggest predictor of adoption delay being socioeconomic status [5]. This implies that the cost of HAs—which is often several thousand dollars—is a significant obstacle to many users. This high cost is partly due to the technology, but also largely due to the closed ecosystem of medical-grade hearing instruments. In response, a new market in off-the-shelf hearing assisted devices has emerged [6] [7].

The National Institutes of Health (NIH) conducted a workshop in 2014 on open-source HA research platforms and published recommendations about their capabilities and features [8]. Our system, **Open Speech Platform (OSP)** [9], is designed to meet these recommendations, including the vision of “new types of basic psychophysical research studies beyond what is widely done today”. OSP is a suite of comprehensive, open-source hardware and software tools for multidisciplinary research in hearing healthcare. The goals of OSP are to address the underlying causes behind the challenges described above, to facilitate existing research by audiologists and DSP engineers, and to enable new kinds of investigations between hearing and related disciplines.

The OSP hardware is comprised of:

1. a Processing and Communication Device (“PCD”), which is a small wearable box containing a smartphone chipset performing all the signal processing and wireless communication functions, plus the battery and supporting hardware
2. “hearing aid”-style audio transducer devices in behind-the-ear receiver-in-canal (“BTE-RIC”) form factor, which connect to the PCD via a 4-wire cable. They support 4 microphones and one receiver (loudspeaker) per ear, plus an accelerometer/gyroscope (IMU) for measuring look direction and researching mobility disorders
3. an optional set of active biopotential electrodes for acquiring EEG or other electrophysiological signals, daisy-chained together and connected to acquisition hardware on the PCD via another 4-wire cable (together called “FM-ExG”)

The OSP software components include:

1. Firmware for FPGAs in the PCD and BTE-RICs
2. An embedded Linux distribution running on the CPU within the PCD, including kernel modifications and custom drivers for the BTE-RICs
3. The OSP real-time master hearing aid (RT-MHA), which is a library of signal processing modules and a reference C++ program that performs basic and advanced HA signal processing in real time
4. The Embedded Web Server (EWS), which:

- a. hosts a WiFi hotspot on the PCD
- b. serves web apps to any browser-enabled device which connects to it, such as the user's smartphone
- c. controls the RT-MHA parameters live based on user actions in the web apps

Taken together, OSP is a powerful research tool, in which all aspects of the assisted hearing experience—from the ear-level hardware to the signal processing algorithms to the way the user interacts with and controls their device—may be customized and used for research in the lab and in the field. The target audience of OSP is not just audiologists and speech DSP engineers, but also researchers in neuroscience, healthy aging, human-computer interaction, networking and edge/cloud processing, wearable electronics, and many other disciplines. Because OSP is open-source—all the software and hardware design files are released on our website [9]—researchers may modify and enhance whatever part of the system is relevant to their work, while leveraging past contributions made by other researchers.

Our development of OSP has resulted in novel developments in embedded systems design [10], portable electrophysiology [11] [12], adaptive filtering [13] [14], and other areas not yet published. Yet, the primary novelty of OSP—and its primary value to the community—is in its system design as a whole, and the capabilities it offers to researchers and users as a result of this design. As such, this paper describes the engineering design of all portions of the OSP platform, with an emphasis on how the design choices provide useful and advanced functionality. In particular, we focus on aspects of the hardware that have not been reported on in previous publications, and we provide updates on continued development of other parts of OSP. Sec. II discusses the PCD, the software from FPGA through kernel level, the BTE-RICs, and other included sensors. Sec. III covers the FM-ExG. Sec. IV reviews the RT-MHA and discusses new academic research on adaptive filters which has already been enabled by OSP. Sec. V describes the software architecture of the embedded web server (EWS) and the current set of provided web apps for audiologist and user engagement. Finally, Sec. VI gives objective performance results for the hardware and software, showing its capacity for real-time, low-latency audio processing, the quality of the recorded electrophysiological signals, and the platform's usability for multidisciplinary clinical research.

A. Related Work

OSP intersects most aspects of the vast field of research on hearing healthcare. Thus, we will restrict our discussion in this section to systems for hearing research that perform real-time audio processing and have a portable or wearable component, as this is what OSP is at its core. The five major HA manufacturers each have their own proprietary systems of this kind, which they use for research on new clinical and technical challenges as they develop their advanced digital HAs. However, these systems are difficult to access for the research community at large, and difficult to modify and to obtain generalizable results from as discussed above.

As of 2014, no non-proprietary HA research system existed which met the needs of the HA research community, according to the aforementioned NIH workshop on this topic: “The

NIDCD-supported research community has a critical need for an open, extensible, and portable device that supports acoustic signal processing in real time” [8]. As a result, in 2016 the NIH awarded six grants for development of open-source hearing aid research tools [15] [16]. Of these six, four—including OSP—are complete master hearing aid tools for research. The other three of these tools are:

a) Tympan: [17] includes a wearable processing unit based on Arduino Teensy [18] and a basic software library for HA processing. The strengths of this platform include flexibility with the transducers (the unit simply features standard 1/8” jacks) and battery (the user selects their own portable battery pack), low cost and use of readily available components, small size, easy development for beginners with the Arduino platform, and fast time-to-market. Its disadvantages include low audio quality, severely limited processing power, and support for only one input channel (microphone).

b) Open-MHA: [19] features an audio expansion board for BeagleBone Black, a Linux-based OS, and an extensive real-time and offline HA software suite. The advantages of this platform include good-quality audio, support for six-channel input, the well-documented nature of both BeagleBone Black and Linux, and the powerful master hearing aid DSP algorithms. Its downsides include somewhat limited processing power, the fact that its form factor is portable but not wearable, and the lack of ear-level transducers for users in the field. However, the open-source nature of these platforms allows the strengths of each to be combined: for instance, the Open-MHA DSP algorithms could in the future be ported to OSP hardware.

c) UT Dallas project: [20] is comprised of a cross-platform smartphone app for processing and commercial Bluetooth-enabled hearing aid transceivers. The advantages of this platform include its advanced speech enhancement algorithms, the complete absence of special-purpose hardware, the accessibility of smartphone development, and the use of industry-standard ear-level transducers (which are proven designs and ultimately the target hardware). Its weaknesses include its inability to process audio in real time (defined as a total microphone-to-loudspeaker delay of less than 10 ms while HA processing is occurring), the proprietary nature of the ear-level transducers, and the semi- or fully-closed smartphone operating systems and driver stack which make it difficult to guarantee performance.

II. WEARABLE HARDWARE

A. Form Factor

As reported in [21], the software portions of OSP were first implemented on a laptop, with a studio audio interface and custom analog hardware for interfacing and the ear-level transducers. The OSP RT-MHA can still run on any Mac or Linux computer using any audio hardware supported by the respective OS. However, the potential of OSP is much more fully realized in its new wearable form factor which we initially discussed in [10].

As discussed in the Introduction, the battery size, available processing power, and communication abilities in commercial HAs are severely limited by the behind-the-ear or in-

ear form factor they typically are available in. These factors in turn contribute to the cost and the difficulty of development (e.g. fixed-point embedded processors). For a research platform, we need much higher processing power, substantially improved wireless communication, relatively low cost, and easy development. These factors are much more important than the entire system fitting behind the ear, so we compromised on the form factor: we created a design which is still easily wearable but which is not limited to the space around the ear (Fig. 1). The processing, wireless communication, and battery for the OSP wearable system are housed in the Processing and Communication Device (PCD), which is a small box that may be worn around the neck or on a belt. The PCD is attached by wires to the BTE-RICs, which contain the audio transducers, codecs and interface hardware, and other sensors. Since the PCD processes the audio from both ears, it can use beamforming and other algorithms to take advantage of binaural information in the audio, something BTE or in-ear HAs would have to use wireless transmission to achieve. The aforementioned NIH workshop suggested that the form factor of BTE-RICs wired to a processing unit would be appropriate for a research system [8].

B. Choice of Embedded Platform

Smartphone chipsets provide best-in-class computational performance per watt, diverse peripherals, and advanced wireless connectivity, so they are a natural choice for the embedded platform in the OSP wearable design. However, many smartphone chipsets are difficult to work with, due to the high degree of proprietary technology in modern smartphones. Furthermore, embedded systems development for hard-real-time, low-latency applications is typically done at a very low level. Low-level audio processing would be contrary to the goals of extensibility and controllability of OSP, but low latency and stability are still mandatory. Thus, the design task was primarily to (1) select a platform which is capable of high-level real-time processing and has all the necessary features, and then (2) adapt its hardware and software to the needs of OSP.

We selected the single-board computer system DragonBoard 410c from Arrow, based on the Snapdragon 410c chipset from Qualcomm. Because of the hobbyist-oriented nature of this product—it is intended to compete with platforms like Raspberry Pi and BeagleBone Black—a large support network for this chipset exists, including a well-maintained Debian branch. Moreover, several companies supply systems-on-module (SoMs) featuring the same chipset, which allow developers to move to an application-specific design without having to design a PCB hosting a complex modern system-on-chip (SoC), while maintaining software compatibility and most hardware compatibility with the DragonBoard. We chose the DART-SD410 from Variscite [22] as our SoM because it breaks out all the multichannel inter-IC sound (MI2S) peripheral lines from the SoC, unlike the DragonBoard and most other SoMs.

The Snapdragon 410c SoC (APQ8016) has four 64-bit ARM A53 cores at 1.2 GHz, plus DSP and GPU. Not only does a multicore CPU provide more processing power than a single-core CPU, it allows us to assign real-time portions of the HA processing to dedicated cores where they will not be interrupted, while the OS and EWS run on a different core (see Sec. II-D). Key SoC peripherals include two multichannel inter-IC sound (MI2S) ports for audio I/O to the behind-the-ear receiver-in-canal (BTE-RIC) transducers; several SPI ports

for peripheral control and communication; a microSD card for data logging; and a UART for the Linux terminal interface. Crucially, the MI2S ports are directly connected to the CPU, unlike in many smartphone chipsets where they are connected to the DSP. The latter would require at least some processing to be done on the DSP, which would substantially complicate the development process compared to running ordinary usermode code on the CPU, or add the additional latency of transfers in each direction. The associated power management IC, PM8916, includes a separate lower-performance codec which is used to provide two microphones on the PCD. The SoC and associated wireless chips provide 2.4 GHz WiFi, Bluetooth, and GPS. Paired with the industry-standard networking software available for Linux, the WiFi can act as an access point and the system can serve web pages to clients which connect to it (Sec. V).

We designed a carrier board to host the SoM (Fig. 3). This board also includes power supplies, the FPGA (Sec. II-F), the other interface hardware and ports for the BTE-RICs and the FM-ExG, the microSD card slot and USB ports, and other basic system features. Adjacent to the carrier board is a 2000 mAh smartphone-type Li-Ion battery, which can be charged from a microUSB port or swapped out by the user. The carrier board, battery, and WiFi antenna are enclosed in a plastic case (Fig. 4) to form the PCD, which may be worn around the neck or on a belt. The PCD is roughly $73 \times 55 \times 20$ mm and has a mass of roughly 83 grams, representing a savings of 67% in weight and 72% in volume over the previous “portable” OSP hardware design [23].

C. Adapting Smartphone SoC Audio Hardware

As discussed above, the 410c platform was chosen for its power efficiency, high performance, wireless capabilities, and product ecosystem. However, the audio subsystem in the Snapdragon 410c was designed to support the needs of low cost smartphones; it was neither designed nor documented for general-purpose use. Our needs for audio I/O to the BTE-RICs in the HA application are substantially different from those of the smartphone applications the SoC’s audio subsystem was designed for. Nevertheless, we were able to adapt this subsystem to the needs of OSP through a combination of reverse engineering and analysis of its partial documentation. Although some of the implementation details discussed here are specific to the 410c SoC, many of them would apply to a variety of single board computers and SoMs based on ARM processors running Linux. The OSP software comprising the RT-MHA and EWS is hardware-agnostic, and can run on Linux and OS X systems in addition to the embedded systems mentioned above.

Specifically, each BTE-RIC has one MI2S data line for microphone data and one for speaker data. The same speaker data line can be sent to both BTE-RICs, with the left and right receiver signals in the left and right time-division slots respectively. However, each of the two microphone data lines must be received by the SoC on separate MI2S data input pins, since they each already contain two mics’ data. This means a total of two MI2S data input lines and one MI2S data output line are needed. Due to the design of the MI2S peripheral units in the SoC and the undocumented multiplexer block which connects them to the SoC’s I/O pins, the only configuration which provides two MI2S data input lines is using two data lines of one MI2S unit in “receive” mode, and using a different unit in “transmit” mode for

the data output line. Unfortunately, the Advanced Linux Sound Architecture (ALSA) kernel subsystem assumes that each codec has a unique data (I2S) port; in our case, two MI2S ports are being shared by two codecs. So, we had to build a custom ALSA driver for the BTE-RICs which registers two “virtual” audio devices—one for mics, and one for speakers—connected to the respective MI2S peripherals. Each virtual device has its own “memory map” with registers controlling the appropriate functions; writes to and reads from these registers are rerouted in the driver to both codecs’ SPI control ports as necessary. The result is that usermode software sees two devices, one with only audio inputs and one with only outputs, both of which function on their own or simultaneously.

D. Embedded Operating System

The embedded operating system used in OSP is based on the Debian 9 (“stretch”) distribution of Linux for Snapdragon 410c (ARM64) provided by Linaro. Besides the custom audio driver mentioned above, we have tailor-built the kernel and configured the environment to meet the following goals:

1. Stable real-time performance
2. Low power consumption
3. Fast bootup
4. Small memory footprint
5. Security

These goals will be referenced by number in the following paragraphs.

Kernel: The kernel is configured with all core facilities and most drivers as built-in. Building as much code into the kernel binary rather than modules improves the bootup time (3). In the current configuration, there are a few drivers that remain modular due to the fact the driver cannot initialize until after the firmware is initialized and the device is powered up. Future work by our team and the community will be needed to modify these drivers to enable them to compile as built-in, which will allow module loading/unloading to be completely disabled. No dynamic loading of kernel-mode code is a desire for security (5) since the kernel cannot be modified as easily at run-time which will help thwart specific threats, e.g., rootkits.

Any kernel facility that is not used by the system or driver that does not have the hardware present is not included. This optimization helps to achieve both (3) and (4) along with the additional benefit of decreasing build time of the kernel. Short build time is not a design goal but is a desirable metric that is crucial in reducing development and test time. Furthermore, we hope to show that by removing additional kernel features, the security posture of the system improves (5) by removing any attack vectors associated with those features.

To address (1), the PREEMPT_RT option has been enabled, which enables the kernel to become preemptable and shortens the critical sections within kernel code.

Environment: systemd has replaced the old sysvinit style init process that becomes PID 1 when the kernel finishes its boot process, and handles the remaining portions of system boot. systemd is configured to only run on CPU core 0 through a configuration setting in `/etc/systemd/system.conf`. As a result of this configuration the init process and subsequent processes spawned by it will only run on core 0. This allows CPU cores 1–3 to be reserved for all real-time processing (1); user code handles their assignment to these cores when they are executed. Similarly, all interrupt handling is bound to core 0 to avoid interrupting the real-time processes running on cores 1-3.

Unnecessary and unused services are disabled to reduce power consumption (2) and enhance system security (5). The Bluetooth radio is also disabled by default for the same two reasons but can be enabled by a user if so desired. As seen in Table III below in Sec. VI, the idle power consumption is more than half of the total power consumption during full operation, so it is extremely important to eliminate unnecessary power sinks to improve battery life.

The system configures the WiFi interface as a hotspot after boot to allow for remote connectivity to the PCD, for the embedded web server (EWS) and for SSH for development. In conjunction with the hotspot, multicast DNS-Service Discovery (mDNS-SD, a.k.a Bonjour) is enabled and configured to allow a user connected to the hotspot to easily access the EWS or SSH into the board using the hostname `ospboard.local`, without needing to know the IP address of the board. As a fallback for systems that do not support mDNS, e.g. Android, the IP address of the board is always the same when connected through the hotspot.

E. High-Performance BTE-RICs

Along with the PCD, the other key hardware component of OSP is novel ear-level transducers in a behind-the-ear receiver-in-canal¹ (BTE-RIC) form factor (Fig. 5). These units are each connected to the PCD via a four-wire cable, and serve as the primary input and output for the system. They are composed of a rigid PCB for the electronics, a flex PCB for the microphones, a custom 3D-printed plastic shell, and a rugged 3D-printed strain relief [9].

Unlike in previous versions, the communication between the BTE-RICs and the PCD is digital—the codecs are within the BTE-RICs. The low-level digital interface is transparently facilitated by FPGAs in both the BTE-RICs and the PCD (Sec. II-F). The decision to have digital communication with the BTE-RICs was made for several reasons. First, analog communication with the BTE-RICs would require at least six wires—a differential pair each for the microphone and receiver, plus power and ground—plus even more wires for multiple microphones per ear. As discussed below, multiple audio inputs per ear is crucial for expansion of the hearing-related research OSP supports. Second, having the codec physically close to the transducers reduces the opportunity for noise and interference. Finally, the digital interface allows for additional sensors at the ear—starting with the IMU (Sec. II-G.1)—without the need for any additional wires, thanks to the FPGAs.

¹The output transducer, i.e. the loudspeaker, is called the “receiver” in the telephony and HA communities. This is typically a small speaker in a long, slender package that is in or just outside the ear canal.

The codec in each BTE-RIC is the high-performance but consumer-grade Analog Devices ADAU1372 [24], which provides a differential headphone driver for the receiver and four analog inputs per ear. By default, these are a front microphone, a rear microphone, an in-ear microphone, and a voice pick-up (VPU) transducer (Fig. 6); while the former two are common on hearing aids, the latter two are for specialized purposes, and are explained below. The I2S standard only supports two channels of audio per data line, so currently only two of these four inputs may be transmitted to the PCD at a time. However, the application may select via ALSA commands which two inputs these are, and future work will enable simultaneous capture of all four microphones (Sec. II-F). All inputs and outputs are sampled at 48 kHz 24 bit; the codec also supports 96 kHz sampling, which will be supported by a future version of OSP for improved accuracy in beamforming.

Several types of audiological studies require measurement of the sound within the ear canal while a hearing assisted device is being worn. Purposes include calibration of the acoustics, Real Ear Insertion Gain (REIG) measurements during HA fitting [25], compensation for occlusion effects [26], and studying otoacoustic emissions [27] [28]. Typically, this measurement is performed with a probe placed into the ear canal as the HA is inserted; unfortunately, this method is time-consuming and precise positioning of the probe can be difficult [25]. To facilitate such studies, the BTE-RICs support a special receiver in development at Sonion [29] which has a microphone in the same package, facing into the ear canal. This allows the sound within the ear canal to be measured and monitored as a normal part of work with the platform—including in the field, which would normally be prohibitively difficult. The current design uses a CS44 connector for the receivers, with a pinout that is compatible with a variety of regular receivers as well as with the embedded-mic receiver, thus not increasing costs for users who do not need this feature.

A VPU (voice pick-up) is a bone conduction transducer that picks up the user's voice, while being highly immune to background noise (40–50 dB loss to ambient sound compared to conducted sound [30]). When mounted to a device which is in robust contact with the head, such as an in-ear hearing assisted device, it picks up the vibrations of the skull—that is, the user's voice—without any outside sound. While bone conduction microphones have made impressive advances, they still have reduced frequency range compared to air microphones, and their response to vibration is noticeably nonlinear. Thus, the VPU in this system effectively provides a measurement of the user's voice which is somewhat distorted but almost completely free of interference. This signal can be useful in several ways. First, adaptive systems such as beamforming and speech enhancement rely on accurate estimates of when the user is speaking (speech presence probability or SPP) in order to estimate the interfering noise. The VPU signal can provide an improved estimate of the SPP, so that the adaptation can be temporarily disabled while the user is speaking [31]. Second, other algorithms can be developed to improve the experience of listening to one's own voice, which is known to be adversely affected by HAs [26]. These may include reducing the gain while the user is speaking, DSP approaches to correct for the presence of the HA in the canal, etc. Finally, algorithms—especially ones involving deep neural networks—can be developed to reconstruct an improved signal of the user's speech from the VPU signal [32], for purposes like telephony or virtual meeting settings.

In addition to the codec and audio hardware, each BTE-RIC also provides an inertial measurement unit (IMU), which is discussed in Sec. II-G.1; separate analog and digital power supplies for additional noise suppression; and an FPGA, which is discussed next.

F. Custom Digital Interface

Both the PCD and each BTE-RIC contain an FPGA (Lattice MachXO3 series [33]). As discussed below, the form factor of the BTE-RICs containing the codecs with processing in the PCD would not be feasible without the FPGAs. Once they were present, they enabled additional features, including the FM-ExG (Sec. III), so they have become a key component of the platform.

The original need for the FPGA came from the observation that the communication between the BTE-RICs and the PCD would require a large number of signal wires: bit clock, word clock, microphone data, and receiver data for I2S, and at least two lines for control signals to and from the codec and IMU (clock and data of I2C). Combined with power and ground wires, the cable to the BTE-RICs would have to have eight conductors. On top of this, neither I2S nor I2C are designed for transmission over wires of any significant length; while they would be likely to work in controlled conditions in the lab, they might not be robust in varying electromagnetic environments in the field. So, we decided to add an FPGA at each end, and transmit all the signals with a custom protocol over a single bidirectional twisted pair, reducing the number of conductors in the cable to four. The physical layer chosen is bus low-voltage differential signaling (BLVDS) [34] [35], a bidirectional version of the popular LVDS standard [36] used in many modern serial interfaces such as USB, SATA, and PCI Express. This interface uses standard CMOS drivers to transmit and analog differential amplifiers to receive; the FPGAs support this interface natively, only needing a few external resistors at each end to match the impedance of the cable. Because the signal transmitted is differential, it is nearly immune to common-mode noise and interference; and since the cable is shielded and the conductors are twisted, there is very little opportunity for differential interference. As a result, this interface is perfect for high-speed communication over the roughly 1 m cable between the BTE-RIC and the PCD.

We created a custom communication protocol over BLVDS, designed to allow the SoC to transparently interact within the codec and IMU within each BTE-RIC (Fig. 7). There are three categories of signals which are multiplexed and packetized for transmission over LVDS: high-speed data, low-speed control, and clock. The microphone and receiver I2S data is the high-speed data; this is transmitted 8 bits at a time in each direction within each communication packet. The SPI control lines for the codec and IMU are the low-speed signals; the states of these signals are transmitted once per packet. Finally, the protocol allows the I2S bit clock in the BTE-RIC to be synchronized with that in the PCD, to correct for drift between the oscillators in the two devices. The FPGA in the BTE-RIC adjusts the sub-cycle timing of its I2S bit clock to match a known rising edge in the data stream from the PCD. Since the BTE-RIC sends back its own rising edge in its half of the packet, each FPGA can determine if the other is connected and properly responding, which allows for deterministic behavior at startup or any time communication is interrupted.

In future work, the FPGAs will also enable simultaneous capture from all four microphones on each BTE-RIC. The codec supports an extension to I2S called TDM which allows for four channels per data line. The SoC's I2S subsystem does not support this, but it does support two channels at twice the sample rate, which is the same data bitrate. For this mode, the FPGA in the PCD will send a "fake" word clock signal to the SoC which matches its expectations and "trick" it into accepting the data. The FPGA will also annotate the channel numbers in the lower, unused bits of the audio data—each sample is 32 bits but the ADC is only 24 bit—so that the application can distinguish them.

G. Simultaneous Ancillary Sensors

As described below, the OSP hardware platform currently supports three additional types of sensing capabilities, not traditionally associated with hearing aids research. Since OSP is designed to be a tool for new kinds of research beyond what is currently possible, these sensors may be used in conjunction with the audio transducers for new work in fields related to hearing, or on their own with OSP acting as a wearable acquisition and processing system. Furthermore, OSP can serve as a baseline open-source wearable hardware design, which can be modified by researchers who would like to add their own sensors for investigations into lifestyle, healthy aging, and many other health-related fields.

1) IMUs in BTE-RICs and PCD: Both the BTE-RICs and the PCD contain a Bosch BMI160 inertial measurement unit (IMU), which is a three-axis accelerometer plus three-axis gyroscope. The gyroscope data from the BTE-RICs provides reasonably accurate information about changes in head orientation. Assuming that target sound sources and interferers move much more slowly or rarely than the user's head, this allows changes in the user's look direction to be corrected for in algorithms which model the spatial positions of audio sources such as beamforming-based source separation or noise suppression. This has the potential to dramatically improve their convergence speed and reduce their error rate, providing a better user experience.

In addition, there is another related healthcare application for the IMU data. Ability to maintain mobility—broadly defined as movement within one's environment—is an essential component of healthy aging, because it underlies many of the functions necessary for independence [37] [38]. In this context, gait disturbances are usually due to a combination of decreased physiological reserves and increased multisystem dysfunction [39]. The IMUs allow researchers to assess gait speed and monitor for unexplained gait disturbances during activities of daily living. Physical activity monitoring software could be developed to run in parallel with the hearing aid software and provide appropriate feedback to the user or researchers.

2) GPS: The SoM includes the radio hardware to support GPS-based location acquisition. Future work will focus on enabling GPS in software and acquiring useful data from it without disrupting real-time audio processing or consuming too much power.

3) FM-ExG hardware in PCD: The PCD's carrier board also includes a hardware subsystem for simultaneous biopotential acquisition. This consists of a fast-sampling ADC

controlled by the on-board FPGA, which relays the data to the Snapdragon SoC via SPI. This system is discussed in the section below.

III. SIMULTANEOUS MULTICHANNEL BIOPOTENTIAL SIGNAL ACQUISITION

A. Background

Acquisition and processing of biopotential or electrophysiological signals—which we call “ExG”, for EEG (electroencephalography), ECG/EKG (electrocardiography), EMG (electromyography), etc.—is a major field of study in emerging healthcare research. Simultaneous EEG and HA audio processing is of particular interest in pre-lingual pediatric hearing loss management, as it could assist clinicians in fitting hearing aids to infants who are unable to self-report the efficacy of their hearing aid prescription, leading to a dramatic improvement in their quality of life [40]. Furthermore, in the future the process of HA tuning could be done adaptively via machine learning systems, which would monitor the experience of the user as measured by their EEG patterns. Unfortunately, EEG typically requires many electrodes with an independent wire for each, making acquisition systems large, expensive, and difficult to use especially in pediatric applications. While devices capable of concurrent hearing aid tuning and EEG do exist [41], to our knowledge no wearable or easily-portable devices of this kind are available to the research community. Other applications of wearable biopotential acquisition systems include monitoring conditions of concern such as heart ailments (ECG), muscle degeneration (EMG), or the progression of neurological disorders (EEG) such as Alzheimer’s disease and Parkinson’s [42]. In addition, there is emerging evidence that neurofeedback from EEG can be helpful as an intervention in many disease conditions [43] including epilepsy [44] and ADHD [45].

B. System Design

OSP incorporates a wearable biopotential acquisition system, which can run alongside the HA processing, and which only requires one small four-wire cable from the electrodes to the PCD. The design of this system is based upon the distributed FM-ADC architecture in [12]. The active electrodes feature high input dynamic range of around 100dB and no input gain stage. This allows them to support wet or dry electrodes, and they can be used for ECG, EMG, and EEG simply by changing the position of the electrodes on the body. In each active electrode, the biopotential signal at baseband is bandwidth-expanded into a frequency-modulated (FM) band centered at a unique carrier frequency. This upconversion is performed in an application-specific integrated circuit (ASIC) and the resultant FM signals are all driven onto a single signal wire, each FM signal occupying a distinct area of spectrum for frequency domain multiplexing (FDM). The electrodes are daisy-chained in any order and connected to the PCD via a 4-wire cable (the remaining three wires being power, ground, and a reference voltage). The aggregate signal content of the single composite FM-FDM wire is sampled by an analog-to-digital converter (ADC) in the PCD. The data can then be streamed using WiFi for off-body processing or processed locally in multi-modal signal processing applications. In either case, after demodulation, the original biopotential signals can be recovered.

The benefits of such a biopotential acquisition system strategy include: power efficiency intrinsic to the distributed FM-ADC architecture, ruggedization against inertial motion artifacts, reduced system weight due to reduced wiring burden, and frequency up-conversion which eliminates baseband coupling artifacts in the signal wire. Its high input dynamic range ensures that the acquisition hardware does not saturate and lose signal for large motion artifacts at the input; combined with the IMUs in the BTE-RICs, OSP could in the future support IMU-based motion artifact removal as demonstrated in [46].

As presented in [11], the FM modulation allows for an increased effective signal-to-noise ratio (SNR_{FM}) compared to the SNR of the ADC at the carrier frequency, called carrier-to-noise ratio (CNR). The overall SNR of the system depends on the bandwidth expansion ratio D [47] as follows:

$$\text{SNR}_{\text{FM}} = 10 \log_{10}\left(\frac{3}{2}D^2\right) + \text{CNR}$$

The CNR of an ideal 12-bit ADC (i.e. 12 effective number of bits or ENoB) is 72 dB, so we chose $D = 20$ to give a theoretical $28 + 72 = 100$ dB SNR for each FM channel. Assuming EEG signals have a maximum frequency of 500 Hz, $D = 20$ leads to a 10 kHz FM frequency deviation. The actual FM bandwidth may be computed two different ways: by the empirical Carson's Rule, giving $2 \times (10\text{kHz} + 500\text{Hz}) = 21$ kHz, or by including all side tones with greater than 1% of the unmodulated carrier amplitude, giving $3.2 \times 10\text{kHz} = 32$ kHz [47]. Based on these two estimates of the bandwidth and the desire for ≈ 10 kHz guard bands between channels, we space the channels 40 kHz apart. With a sampling frequency of 1 MHz, 12 ExG channels can be supported.

An overview of the hardware included on the PDC to realize this is shown in Fig. 8. The Analog Devices AD9235 [48] was chosen for its parallel interface, 12-bit resolution, and supported sampling rates up to 60 MHz. The ADC is clocked by the FPGA with a 1.024 MHz clock signal generated by dividing the 12.288 MHz clock from the MEMS oscillator driving the I2S by 12. The ADC's parallel data interface connects to the FPGA, which contains a simple FIFO queue to store the samples until they are ready to be retrieved by the SoC via SPI. A level-based signal is sent to the SoC when more than 1024 samples (1 ms of data) are available; the SoC polls this signal and then performs an SPI transfer of 1536 bytes, which covers the 1024 12-bit samples. Since the SPI clock runs at 50 MHz—which could theoretically transfer 6250 bytes per ms if the clock ran continuously—there is sufficient timing slack for transfers to be stable.

When FM-ExG streaming is running, CPU core 3 is dedicated to the FM-ExG thread. It runs at the highest real-time priority and is the only thread permitted to run on this core. It polls the “data ready” signal described above, performs the SPI transfers, and executes a callback to user code for each 1 ms (1024 samples) of data received. Any processing or transmission of the data for any research application would occur during this callback. We created two programs which implement this callback to collect results as described in Sec. VI-C: one which measures the time between rising edges of a pulse wave for the sync measurements,

and one which saves 10 seconds of data to RAM and then to disk. In the latter case, we performed digital demodulation offline using MATLAB.

C. Future Work

Our first goal for future work with FM-ExG is to enable demodulated data to be streamed via WiFi from the PCD. This will require creating a real-time implementation of the demodulator, ensuring its performance is high enough to run in the callback without disrupting the data capture, and implementing both the local and remote side of the WiFi streaming system. Once this is accomplished, we are excited to begin exploring clinical uses of FM-ExG, particularly in pediatric hearing loss research.

IV. REAL-TIME MASTER HEARING AID (RT-MHA)

A. Baseline Algorithms

We provide a full set of baseline implementations of common HA algorithms in the RT-MHA, to facilitate basic HA research with the platform and to provide a reference implementation for engineers to build from. An overview of the RT-MHA signal flow is shown in Fig. 9. These algorithms are essential components of any HA, and can be categorized into “basic” and “advanced” functions. The basic HA functions necessary for amplification are:

1. Subband decomposition
2. Wide dynamic range compression (WDRC)
3. Adaptive feedback cancellation (AFC)

Many commercial HAs include advanced features to improve speech perception in realistic situations such as in a noisy environment. The RT-MHA implements two advanced functions for improving conversation in noise: resume

1. Speech enhancement (SE)
2. Microphone array processing (or beamforming)

In the following we briefly describe the role and our baseline implementation of each of these five algorithms.

1) Subband decomposition: Hearing loss is typically highly frequency dependent; it is common for loss to be worse at high frequencies, but loss curves vary widely among individuals. Hence, gain and other processing must be applied differently at different frequencies, motivating the decomposition of the input signal into frequency bands. In the RT-MHA, this decomposition is implemented as a bank of 6 finite impulse response (FIR) filters, where the bandwidths and upper and lower cutoff frequencies of these filters are based on Kates’s MATLAB master hearing aid implementation [49].

2) WDRC: Both healthy hearing and hearing loss are known to be nonlinear in amplitude, with these nonlinearities varying over frequency. Therefore, a gain control mechanism that enables a frequency-dependent, nonlinear gain adjustment is needed for modern HAs. This

is carried out by the wide dynamic range compressor (WDRC), which is one of the essential building blocks of a HA [50]. The WDRC amplifies soft sounds while limiting the gain of loud sounds, with the aim of improving audibility without introducing discomfort. Typically, WDRC amplifies quiet sounds (40-50 dB SPL), attenuates loud sounds (85-100 dB SPL), and applies a variable gain for everything in between. The basic WDRC system described in [51] comprises an envelope detector for estimating the input signal power and a compression rule to realize nonlinear amplification based on the estimated power level. Primary control parameters of the basic WDRC system are: attack time (AT), release time (RT), compression ratio (CR), gain at 65 dB input (G₆₅), and upper and lower kneepoints (K_{up} and K_{low}) [51]. The AT or RT is the time the envelope detector takes to recover the output signal level to its steady state when a sudden rise or drop takes place in the input signal level, respectively. The amount of gain to apply will then be determined based on a compression rule as a function of the estimated input power level given by the envelope detector. The CR, G₆₅, AT, RT, K_{up}, and K_{low} are the control parameters for characterizing the compression rule. In the RT-MHA, the above WDRC is implemented in a 6-channel system [51], where gain control is realized independently in each subband, enabled by selecting different parameters to specify the compression rule. The outputs of all the subbands after applying the WDRC are combined together to produce the HA output signal.

3) AFC: Feedback due to acoustic coupling between the microphone and receiver is a very well-known problem in HAs [51]. There are many methods to alleviate this phenomenon [52]. Among them, adaptive feedback cancellation (AFC) has become the most common technique because of its ability to track the variations in the acoustic feedback path and cancel the feedback signal accordingly. The AFC generates an estimate of the feedback path using an adaptive finite impulse response (FIR) filter that continuously adjusts its filter coefficients to emulate the feedback path impulse response. Typically the AFC can provide 5-12 dB added stable gain (ASG) [14] depending on the adaptive filtering algorithms used. The RT-MHA implements the least mean square (LMS) based algorithms and features the sparsity promoting LMS (SLMS) [13] which is an advanced adaptive filtering algorithm developed by the OSP team and discussed below (Sec. IV-B).

4) SE: In a quiet environment, the above features of HAs are enough to help the user better understand speech. However, in a noisy environment such as a cafeteria or a restaurant, the HA might not be able to improve conversations without any noise reduction mechanism—for example, WDRC may amplify noise components along with soft sounds. It is therefore essential to have reliable and robust speech enhancement (SE) systems implemented in the HA. A baseline SE module, based on a version of the SE systems investigated in [53], has been added to the RT-MHA. The SE module performs denoising in the subband domain, between the subband decomposition and the WDRC blocks.

5) Microphone array processing: To improve speech intelligibility in noisy environments, RT-MHA implements a baseline left/right two-microphone adaptive beamforming (BF) system. This baseline system described in [54] realizes the generalized sidelobe canceller (GSC) implementation [55] of the linearly constrained minimum variance (LCMV) beam-former [56]. Fig. 10 depicts the BF block diagram. For the adaptation, an

adaptive filter using the (modified) LMS [57] is used to continuously estimate the interference signal components. In addition, adaptation-mode-control and norm-constrained adaptation schemes have also been incorporated to improve robustness [58], i.e., to mitigate misadjustment of the BF due to array misalignment, head movement and shadow effect, room reverberation, etc. Based on simulation with one target and one interference speech signal, the baseline 2-mic beamformer improves the Signal-to-Interference Ratio (SIR) from 1.6 dB to 15.8 dB, and the Hearing-Aid Speech Quality Index (HASQI) from 0.21 to 0.43 over the system with only one microphone (i.e., no beamformer). In informal subjective assessments, the listeners were given a web app for turning the beamforming on/off. All listeners reported a perceived reduction in the interfering speech and background noise with beamforming enabled.

B. Case Study: SLMS

One of the purposes of OSP is to provide a platform for academic research in DSP with easy prototyping, high-quality real-time I/O, and a strong connection to the clinical research community. As an example of such research already performed with this platform, we briefly describe the sparsity promoting LMS (SLMS) algorithm [13] used in several of the adaptive filters on the platform. The SLMS is an adaptive filtering algorithm that takes advantage of the sparsity of the underlying system response—which is present in many HA DSP applications—for improved convergence behavior when adapting the filter coefficients. In testing on early versions of OSP, we have found the SLMS to be useful in the AFC and the adaptive beamforming subsystems. In the AFC, typical feedback path impulse responses are (quasi) sparse in nature, which means they contain many zero or near-zero coefficients and few large ones. It has been shown in [13] that a proper p value of the SLMS parameter leads to a performance improvement. We reported 5 dB improvement in added stable gain with a p of 1.5 for the SLMS over the conventional methods. For adaptive beamforming, the two-microphone GSC system of [54] also benefits from using the SLMS for the filter coefficient adaptation. We have found that improvement in signal-to-interference ratio (SIR) can be achieved for a p of 1.3 ~ 1.5. For reference, $p = 1$ in SLMS results in the ℓ_1 norm similarly used in the well-known proportionate normalized LMS (PNLMS) [59] and $p = 2$ results in the ℓ_2 norm which yields the standard LMS.

V. EMBEDDED WEB SERVER

Most commercial HAs provide smartphone apps for the user to control various aspects of their HA. Recent evidence suggests that adults with hearing loss who have access to smartphone-based tools feel more empowered, autonomous, and in control of their hearing loss [60]. While smartphone apps hold much promise for both professionals and patients, a significant amount of research is needed in terms of assessment and guidance for informed, aware, and safe adoption of such apps by the community [61]. In order to fulfill the visions of the NIH workshop [8], we undertook development of multiple classes of such apps aimed at users (people with HL controlling their HAs), researchers (clinicians engaged in hearing healthcare research and translation), and engineers (those contributing to OSP and the open source initiative).

Most modern mobile-oriented applications fall into two categories: native apps and web apps. Web apps would typically require a remote server and guaranteed availability of an Internet connection, and thus be unsuitable for a wearable system to be used in the field. However, due to the processing power and wireless connectivity of the Snapdragon 410c SoC and the well-developed web software infrastructure on Linux, we are able to host a WiFi hotspot and a web server directly on the PCD. Thus, any browser-enabled device (such as a smartphone or a tablet) can connect to the PCD without the need for any external hardware or connection. As a result, the design decision of native apps versus web apps remained. Native apps can have better hardware integration and certain aspects of user experience, while web apps have the benefits that they do not require installation, they are operating system and form-factor agnostic, and they are easier for programmers to modify and extend [62]. For these reasons and especially due to the ability to rapidly prototype with web apps, we adopted web apps and developed the Embedded Web Server (EWS) subsystem of OSP to support them. All together, the EWS comprises (i) a WiFi hotspot for browser-enabled devices to connect to, (ii) a web server running on the PCD, (iii) bidirectional communication between the web server and the RT-MHA for monitoring and control, and (iv) a suite of web apps hosted on the web server. Researchers can customize these apps to enable a broader range of research in hearing healthcare.

A. EWS Architecture / Software Stack

The EWS on OSP is implemented using the LAMP stack (Linux OS, Apache web server, MySQL database, and PHP scripting language) [63]. The web apps themselves are coded using HTML, CSS, and JavaScript. We have chosen SQLite as the database and a test server provided by the PHP framework as the web server. The choice of SQLite and PHP test server were guided by the fact that they do not require complex configuration steps like Apache and MySQL do. In addition, they are very lightweight from processing load and memory footprint perspectives. In the context of realtime monitoring and control of RT-MHA from a browser enabled device, we have a limited number of connections and many of the features of Apache and MySQL are not relevant.

The RT-MHA serializes OSP parameters between a binary representation in memory and a JSON string format for communication with EWS over a TCP/IP socket. All the RT-MHA parameter states are stored in the SQLite database for persistency and use by the web apps.

B. Web Apps

In order to expedite web app development, OSP provides Laravel and [Node.js](#) frameworks. Web apps in OSP present a graphical user interface (GUI) to the user via their device's browser. Based on the user's interactions with the GUI, the apps' control logic may modify the RT-MHA parameters, play back audio to the user through the BTE-RICs, record audio from the microphones, store information in the SQLite database or in logs, or take other actions. In this section, we describe the current suite of web apps, which showcase the functionality of the EWS and OSP as a whole and which serve as templates to be modified and extended for specific investigations.

1) Researcher App: The “Researcher App” is used to manipulate any of the exposed RT-MHA parameters. The main tab of this app includes all the WDRC parameters in each subband. Researchers can save different configurations in named files and load them from the GUI. A Transmit button sets the RT-MHA to the parameters displayed in the GUI. The researcher can individually control the right ear channel or the left ear channel, or both at the same time. The Noise Management tab has the parameters associated with noise management algorithms described in Secs. IV-A.4 and IV-A.5. It enables researchers to experiment with various parameters and provide configurations such as aggressive, mild and no noise suppression in studies with human subjects. Similarly, the Feedback Management tab allows the researcher to optimize AFC parameters for specific investigations. This app is suitable for “audiologist fit” research by entering the user’s initial prescription from a fitting such as NAL-NL2, DSL, etc. [64] and then optimising the HA for user comfort. The researcher app, like the other apps, requires a researcher ID and user ID to access, allowing user profiles to be easily loaded for clinical studies in which one system is used sequentially by multiple users.

2) Self-Fitting Apps: There has been a lot of interest in self-fitting research, wherein the user is able to choose the HA parameters with the help of apps. The recent passage of the Over-the-Counter (OTC) Hearing Aid Act of 2016 was aimed at easing the financial burden of owning HAs, at least for some users with mild to moderate hearing loss. The use of OTC HAs will require users to be able to independently control the HAs in multiple listening environments without professional assistance.

We have implemented baseline web apps for two self-fitting paradigms. First, for the lab based OSP system [21], we initially implemented a native Android version of the Goldilocks explore-and-select self fitting protocol proposed in [65] [66]. For the wearable system [10] aimed at field studies, we transitioned to web apps for the ease of rapid prototyping and ported Goldilocks as a web app.

Second, we created an AB app in which the user can switch between hearing an A or B set of RT-MHA parameters for the same stimulus, and then select the one that they prefer. For the baseline implementation, the app performs a binary search over the overall gain parameter, allowing the user to narrow in on the gain they most prefer. This is intended as a proof-of-concept for researchers to incorporate other HA parameters in their self-fitting research.

This AB app, like several others described below, relies on the audio file I/O module included in OSP. This module, under control of the EWS, can play audio files (typically speech content) stored on the PCD to the user, with or without the RT-MHA processing. This capability allows researchers to provide stimuli to the user in a repeatable and reproducible manner. The file I/O module can also record the raw or processed microphone audio to audio files on the PCD, as described below.

3) Monitoring user and environment state: We have created an Ecological Momentary Assessment (EMA) web app, which is designed to help researchers understand more about the user’s actions in the context of an experiment or a self-fitting adjustment. It

does so by collecting information about the environmental state that elicited the user's behavior along with the user's behavior itself. The EMA web app has two components. First, it displays a brief survey through the GUI which asks the user qualitative questions about their experience and environment. Researchers can edit the survey questions by changing the contents of the JSON file associated with the EMA web app. Second, the app records microphone audio in order to characterize the user's auditory environment. This works with a circular buffer that temporarily keeps the last few seconds of microphone audio. When the EMA is started, the previous buffer is saved, and the audio continues to be saved while the user completes the survey and for a certain time after leaving the app. In the future, the information gathered from the EMA web app could be used to create machine learning models that dynamically update their parameters depending on environmental factors.

4) Outcomes assessment: This class of apps is aimed at assessing the benefits to the user of a proposed hearing loss intervention (such as a particular fitting or an entire self-fitting paradigm). In these apps, researchers define a series of questions in which the user hears pre-recorded sound stimuli (typically speech) and indicates their preference among them or attempts to distinguish between them. The stimuli are processed through specific HA parameter sets during playback, so they can be used to assess the effectiveness of these fitting parameters for the user. The environment audio recording described above may also optionally be enabled in these apps.

In the 4-Alternative Forced Choice (4AFC) app (Fig. 11), each question has a playable prompt stimulus and four written words, one of which matches the stimulus. The words are themselves also playable, and any errors in the user's choices can inform the researcher about what improvements may be needed in the user's HA fitting. The app can easily be modified to create N -alternative forced choice tests.

In the outcomes assessment AB app, the user hears two different stimuli A and B, and rates their preference for B relative to A on a Likert scale. At the researcher's option, A and B may be different audio files played through the same set of RT-MHA parameters, or the same audio played through different parameter sets. In the latter case, the audio may be from a file or it may be the live real-world sound from the user's environment.

Finally, in the ABX app, the user is presented with a target stimulus X, and then two stimuli A and B where one is identical to X and the other is typically very similar. The user selects the one they believe is identical; errors imply that the user could not hear the difference between A and B. This approach has strong discriminative power; its uses include optimizing signal processing (for example, whether the user can detect distortions introduced by approximate computations to save battery power), determining just noticeable differences between parameter settings, etc.

C. Web App Customization

The current suite of web apps are meant to function as baseline, reference implementations for the development of new web apps. Some web apps can be reconfigured for new users and new trials by the researchers without modifying the software. For example, in the case of the outcomes assessment web apps (Sec. V-B.4), the researchers can specify the contents

of the questions which will be shown to the user. In the case of 4AFC, for one question, the researcher needs to specify the audio file for the prompt, as well as the text and audio files of the four choices. The researcher can encode these choices by editing the text-based JSON file that accompanies the app. The audio files themselves are stored in a specific hierarchical file structure, so that a researcher can easily track which files are associated with which question, and have a consistent scheme to document the files referenced in the JSON file. Similarly, the AB and ABX web apps also have JSON files that are used to specify which sound files should be played for which question, which can also be edited with a text editor.

It is also possible to combine aspects of different web apps to create new apps for novel investigations. This requires familiarity with HTML, JavaScript, and PHP. When new HA parameters are exposed by the RT-MHA signal processing, they can also be easily integrated in the web apps with appropriate changes to the HTML and JavaScript (for the modified GUI) and the PHP (for the HA parameter control logic).

VI. RESULTS

Initial results about the performance of the wearable OSP system were reported in [10]. This section summarizes those results and includes updated results based on the current internal development versions of the OSP hardware and software (a version of which will become Release 2019b). In addition, the results relating to the FM-ExG are reported here for the first time.

A. HA Performance

1) Latency: Latency plays an important role in users' comfort with their devices [3] [4], and most commercial HAs have under 10 ms latency [67]. In the OSP wearable system, with the RT-MHA algorithms disabled and the software set to simply pass through an amplified copy of the front microphone input signal to each receiver, the microphone-to-loudspeaker latency is about 2.4 ms. This delay is caused by input and output buffers of 1 ms each allocated by the audio subsystem (ALSA / PortAudio), plus additional delays due to resampling filters within the codec. With the RT-MHA enabled without beamforming, it measures at about 4.6 ms, with this 2.2 ms difference being due to the FIR filters within the audio processing. With beamforming enabled and set to 5 ms delay, the latency is measured to be 9.6 ms as expected. Thus, our full-featured baseline implementation meets the 10 ms target maximum latency for a HA system. While the latency due to hardware and firmware (2.4 ms) is not user-adjustable, the latencies due to the steps of HA processing are determined by the parameters of that processing (e.g. the length of the FIR filters), which will vary as researchers tweak the baseline algorithms and implement their own. The latency "budget" of 7.6 ms for all HA processing allows for a wide range of experimentation and research.

2) ANSI 3.22 Test Results: ANSI 3.22 [68] is a standard test protocol for HAs, the results of which are available for commercial HAs. We measured the OSP wearable system, as well as the previous OSP laptop-based system, with the Audioscan Verifit 2 test unit [69], for comparison with four anonymous commercial HAs.

The OSP wearable system meets or exceeds the performance of the commercial HAs on most metrics. With the high-power (bandwidth-limited) receiver, it provides higher OSPL90 (loudness) with gain, bandwidth, noise, and distortion figures which are comparable to the best of the commercial HAs. With the high-bandwidth receiver, it has similar performance with slightly reduced gain, but with higher distortion. Reducing the gain from 35 to 25 dB (not shown in the table) did reduce the distortion to 1 % or less in all bands. We believe this distortion is due to impedance differences between the two receivers: the codec's output voltage swing is limited by its 3.3V supply rail, which will lead to distortion at a lower power with the higher-impedance (high-bandwidth) [70] receiver than with the lower-impedance (high-power) [71] receiver. Future BTE-RIC designs could add a boost regulator and additional power amplifier to increase the gain with the high-bandwidth receiver; nevertheless, the OSP wearable system meets its performance goals with the high-power receiver.

B. Embedded Software Performance

1) CPU Usage: Each audio channel (left and right ear) is processed by a separate thread so most of the computation can be done simultaneously on two CPU cores. Three of the four cores are assigned to the RT-MHA process at the OS level with the remaining core left for all OS functions and other non-realtime processes. The RT-MHA process is also given maximum CPU and I/O priority. The RT-MHA processes audio in 1 ms frames (48 samples), which means the system has less than 1 ms of real time to complete the processing of each frame. Thus, we report the real time required for each step of the RT-MHA processing.

As shown in Table II, on average the processing completes with some time to spare. In addition, while most of the processing is being done by two cores (one per ear), there are three cores available for the RT-MHA as long as the FM-ExG is not in use. In this case, a substantial amount of additional processing could be added on a third thread provided that it could be done in parallel. Between 2018c and the current version, the subband and AFC filter lengths were reduced somewhat to free up CPU and latency budget for the addition of beamforming. In addition, the “maximum time” values, which effectively measure the algorithms' stability in terms of CPU usage, have dramatically reduced. This is partly due to stability and performance improvements in the OS, and partly due to improved initialization in the RT-MHA. However, it is also partly an artefact of measurement changes: in 2018c, we measured average and maximum times beginning as soon as the RT-MHA started, whereas now we begin timing measurements after the RT-MHA has initialized and run for about a second. Thus, previously the “maximum times” included initialization, whereas now they only measure timing variation in steady state.

2) Battery Life: We measured the current draw of the wearable system in several conditions, and computed the battery life from these assuming a 2000 mAh battery:

As seen in Table III, both system and RT-MHA efficiency have been improved since last reported. Note that due to battery capacity not being fully exhaustible and other factors that may make the usable energy of a battery less than its rated capacity, actual usage times may

be lower than reported here. Still, these results indicate that the system should provide at least 4 hours of full-featured operation per charge.

C. FM-ExG Performance

1) Simultaneous Acquisition: Several metrics are important in characterizing the performance of simultaneous audio and FM-ExG capture on the OSP hardware/software platform. First is the stability of the simultaneous capture—how frequently data is lost in either stream while the system is streaming them both. We tested this by running simultaneous capture from both streams into a simple utility which validated whether and when samples were lost. Over a 90-minute test, no samples were lost on FM-ExG (about 5.5 billion consecutive samples received correctly), and only one incident occurred where a few ms of audio samples were lost. Second is the long-term drift or relative inaccuracy in the sample rate of both streams. This is guaranteed to be zero by design: both the 1.024 MHz FM sample clock and all the audio clocks are derived from the same 12.288 MHz MEMS oscillator which drives the FPGA, so any drift or inaccuracy in this oscillator will be reflected uniformly in the two data streams.

Finally, a key metric for simultaneous capture is how closely the two streams can be synchronized in time. We use the term *skew* to refer to the time difference between the audio and EEG sampled data streams. Since any known skew can be corrected for by simply re-aligning the two data streams, the metric of interest is the variability of the skew over different runs. To help synchronize the two streams, we created a “sync” feature in the OSP FM-ExG API, which signals the FPGA to insert about 1 ms worth of zeros into both the FM-ExG data stream and all microphone audio data streams. Then, a utility detects this period of exactly zero data (which is virtually impossible to occur naturally due to system noise) and marks the end of this period as corresponding to the same time in both streams. To determine the remaining skew and the skew variability after this offset was corrected for, we used a signal generator to input a pulse wave into both the FM-ExG and microphone inputs, and in software measured the timing of the rising edges relative to the sync zeros period. Fig. 12 shows the resulting skew over 32 trials.

The FM-ExG signal path should theoretically have a delay of about 4-5 samples due to the pipelined ADC, which accounts for about 4-5 μs ; the audio signal path should have a delay of about 4-8 samples due to the resampling filters in the codec, which accounts for 80-160 μs . The measured average skew of 135 μs meets our expectations. More importantly, the standard deviation of the skew is about half the audio sample period; of course, it is not possible to identify the timing of a step signal from a sampled representation with better precision than the sample period. Since this uncertainty holds for both the sync zeros period and the pulse wave edges, and the subsample positioning of each of these are presumably uncorrelated, we expect a spread of about $\sqrt{2}$ times the audio sample period, which closely matches the data. Hence we claim that OSP allows for simultaneous FM-ExG and audio streaming synchronized to within about 2 audio sample periods, or about 40 μs .

2) Analog Performance: To evaluate the analog performance of the FM-ExG signal acquisition system, a 100Hz test sine wave modulating a 250 kHz center frequency FM

carrier with bandwidth expansion of 20 (to fit the FM bandplan outlined in Sec. III-B) was generated by MATLAB's `fmmod()` function and driven into the FM-ExG ADC introduced in Section III.B. by a National Instruments USB-6361 DAQ Multifunction Analog/Digital I/O Device. After being sampled and recorded by the PCD, the data was copied to a computer where it was demodulated using MATLAB's `fmdemod()` to recover the original test signal. Fig. 13 depicts the frequency domain representation of the result of this process, which demonstrates 94dB of signal-to-noise ratio (SNR). Compare this to the theoretical 100dB described in Sec. III-B. While this is more than enough SNR for the target application, we believe this measurement may have been partially limited by the test equipment. The DAQ test device mentioned above has a timing resolution of 10 ns, which limits the precision with which the FM carrier wave's instantaneous frequency could be generated, and thus the resolution with which the message signal could be encoded on the carrier wave.

D. Results Summary

OSP meets or exceeds the performance of four representative commercial HAs on the ANSI 3.22 test protocol with an appropriate receiver. OSP also matches the latency of commercial systems with its baseline algorithms (< 10 ms), although its latency will vary as researchers reconfigure it with optimized or additional algorithms. Its capabilities for wireless control, monitoring, and user interaction via the EWS enable rapid prototyping for clinical investigations that may not be possible with most commercial systems. The CPU occupancy reported in Table II and current draw reported in Table III are only partially optimized, and may be improved further by the open-source community. The addition of 6 DOF IMUs at ear level and the capability of acquiring multichannel EEG synchronized with auditory stimuli with about $40 \mu\text{s}$ are expected to facilitate psychophysical investigations beyond what is currently possible. In conclusion, OSP meets the requirements of the community as a HA research platform; it is not a form-factor-accurate HA, in the sense of commercial HAs.

VII. CONCLUSION

Open Speech Platform (OSP) is a comprehensive hardware and software platform for research in hearing healthcare and related fields. It is designed to facilitate lab and field studies in speech processing algorithms, human sound perception, HA fitting procedures, and much more, while also enabling new kinds of research which were never before possible.

The OSP PCD hardware contains the quad-core Snapdragon 410c smartphone chipset running a custom-optimized Debian Linux OS. The PCD software comprises basic and baseline advanced binaural HA audio processing algorithms, which run in real time with CPU resources to spare. The total microphone-to-loudspeaker latency due to hardware and OS is about 2.4 ms. Currently, basic HA processing adds 2.2 ms of latency and beamforming adds an additional 5 ms, for a total latency of 9.6 ms. The PCD is packaged in a small, light plastic case, roughly $73 \times 55 \times 20$ mm with a mass of roughly 83 grams. It contains enough battery power for at least 4 hours of operation with all features enabled.

OSP includes custom ear-level transducers in BTE-RIC form factor. They support up to four microphones per ear, including special-purpose in-ear and VPU microphones, and sample all

inputs and outputs at 48 kHz 24 bit with hardware support for 96 kHz. They also contain an six-axis IMU for measuring look direction, assessing balance, and other physical activity research. The BTE-RICs communicate with the PCD via a custom packetized protocol over LVDS facilitated by FPGAs at either end, which transmits high-speed audio, control, and clock information over a single differential pair in a thin four-wire cable.

The OSP PCD is also the gateway for FM-ExG, a low-power wearable biopotential signal acquisition system for collecting EEG, ECG/EKG and EMG signals. The PCD includes a high-speed ADC and interface logic in the FPGA, to enable acquisition of 12 channels of biopotential signals with a measured SNR of 94 dB. FM-ExG can run while the HA processing is occurring, for simultaneous acquisition of audio and EEG synchronized to within 40 μ s and with no long-term drift.

Finally, the PCD hosts a WiFi hotspot and web server which users and researchers can connect to with any browser-enabled device. The OSP software framework serves web apps from the PCD which allow users to interact with the parameters of the HA processing in real time. The web apps provided with the current release of OSP include apps for direct monitoring and control of all HA parameters, self-fitting, collecting data about the user's environment, and assessing HA performance. The web apps use a popular software stack and are easy to modify and extend, so that researchers can adapt them or design new web apps to conduct novel studies and field trials.

OSP has been architected to fulfill the vision set out by the NIH workshop [8] for an open, extensible research tool for hearing healthcare and related fields. OSP meets all of the basic requirements presented there—portable hardware, real-time signal processing, advanced processing power, wireless controllability, a reference HA implementation, and open-source hardware and software releases. It further meets many of the advanced or optional suggestions: wearability, use of an FPGA in the signal chain, binaural processing, and incorporation of sensing paradigms not traditionally associated with hearing aids, such as FM-ExG and the IMUs. OSP is a powerful set of tools which promote the open initiative for collaborative work on research hardware and software, towards new discoveries in hearing-related healthcare research.

Acknowledgments

This work is supported by the National Institute of Health, NIH/NIDCD grants R21DC015046, R33DC015046, "Self-fitting of Amplification: Methodology and Candidacy," and R01DC015436, "A Real-time, Open, Portable, Extensible Speech Lab," awards to University of California, San Diego; by the U.S. Army Research Laboratory under Contract W911QX-16-C-0003; and by the National Science Foundation Graduate Research Fellowship under Grant DGE-1144086, and grant IIS-1838830 from the Division of Information & Intelligent Systems, "A Framework for Optimizing Hearing Aids In Situ Based on Patient Feedback, Auditory Context, and Audiologist Input."

Support from Sonion for providing emerging ear level transducers (in-ear microphone integrated in the speaker module and "VPU" bone conduction microphone), traditional BTE-RIC transducers, and other electromechanical components is greatly appreciated.

REFERENCES

- [1]. Bennett RJ, Laplante-Lévesque A, Meyer CJ, and Eikelboom RH, “Exploring hearing aid problems: perspectives of hearing aid owners and clinicians,” *Ear and hearing*, vol. 39, no. 1, pp. 172–187, 2018. [PubMed: 28787315]
- [2]. American Speech-Language-Hearing Association et al., “Guidelines for manual pure-tone threshold audiometry,” tech. rep., ASHA, 2005.
- [3]. Agnew J and Thornton JM, “Just noticeable and objectionable group delays in digital hearing aids,” *Journal of the American Academy of Audiology*, vol. 11, no. 6, pp. 330–336, 2000. [PubMed: 10858005]
- [4]. Stone MA and Moore BC, “Tolerable hearing aid delays. I. estimation of limits imposed by the auditory path alone using simulated hearing losses,” *Ear and Hearing*, vol. 20, no. 3, pp. 182–192, 1999. [PubMed: 10386846]
- [5]. Simpson AN, Matthews LJ, Cassarly C, and Dubno JR, “Time from hearing aid candidacy to hearing aid adoption: a longitudinal cohort study,” *Ear and hearing*, vol. 40, no. 3, pp. 468–476, 2019. [PubMed: 30085938]
- [6]. Kochkin S, “MarkeTrak VIII: utilization of PSAPs and direct-mail hearing aids by people with hearing impairment,” *Hearing Review*, vol. 17, no. 6, pp. 12–16, 2010.
- [7]. Brody L, Wu Y-H, and Stangl E, “A comparison of personal sound amplification products and hearing aids in ecologically relevant test environments,” *American journal of audiology*, vol. 27, no. 4, pp. 581–593, 2018. [PubMed: 30458521]
- [8]. Miller RL and Donahue A, “Open speech signal processing platform workshop,” tech. rep., National Institutes of Health, Bethesda, MD, 10 2014.
- [9]. THE_Lab at UC San Diego, “Open Speech Platform.” <http://openspeechplatform.ucsd.edu/>, 2019.
- [10]. Pisha L, Hamilton S, Sengupta D, Lee C-H, Vastare KC, et al., “A wearable platform for research in augmented hearing,” 2018 52nd Asilomar Conference on Signals, Systems, and Computers, no. 52, 2018.
- [11]. Warchall J, Kaleru S, Jayapalan N, Nayak B, Garudadri H, and Mercier PP, “A 678 uW frequency-modulation-based ADC with 104 dB dynamic range in 44 kHz bandwidth,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 10, pp. 1370–1374, 2018. [PubMed: 31285711]
- [12]. Warchall J, Theilmann P, Ouyang Y, Gamdadri H, and Mercier PP, “A rugged wearable modular ExG platform employing a distributed scalable multi-channel FM-ADC achieving 101dB input dynamic range and motion-artifact resilience,” 2019 IEEE International Solid-State Circuits Conference - (ISSCC), pp. 362–363, 2019.
- [13]. Lee C-H, Rao BD, and Garudadri H, “Sparsity promoting LMS for adaptive feedback cancellation,” in *Proc. Europ. Signal Process. Conf. (EUSIPCO)*, pp. 226–230, 2017.
- [14]. Lee C-H, Kates JM, Rao BD, and Garudadri H, “Speech quality and stable gain trade-offs in adaptive feedback cancellation for hearing aids,” *J. Acoust. Soc. Am*, vol. 142, no. 4, pp. EL388–EL394, 2017. [PubMed: 29092590]
- [15]. Miller RL, “Open design tools for speech signal processing (R01),” tech. rep., National Institutes of Health and National Institute on Deafness and Other Communication Disorders, 2015.
- [16]. Miller RL, “Open design tools for speech signal processing (R43/R44),” tech. rep., National Institutes of Health and National Institute on Deafness and Other Communication Disorders, 2015.
- [17]. Clavier O, Audette C, Rasetshwane D, Neely S, et al., “Tympan.” <https://tympan.org/>, 2019.
- [18]. Stoffregen P, “Teensy USB development board.” <https://www.pjrc.com/teensy/>, 2019.
- [19]. Obbard C, James D, Herzke T, Kayser H, et al., “Open community platform for hearing aid algorithm research.” <http://www.openmha.org/>, 2019.
- [20]. Panahi I, Kehtarnavaz N, Thibodeau L, et al., “Smartphone-based open research platform for hearing improvement studies.” <https://www.utdallas.edu/ssprl/hearing-aid-project/>, 2019.
- [21]. Garudadri H, Boothroyd A, Lee C-H, Gadiyaram S, Bell J, Sengupta D, Hamilton S, Vastare KC, Gupta R, and Rao BD, “A realtime, open-source speech-processing platform for research in

- hearing loss compensation,” in 2017 51st Asilomar Conference on Signals, Systems, and Computers, pp. 1900–1904, IEEE, 2018.
- [22]. Variscite, “DART-SD410: Qualcomm Snapdragon 410.” <https://www.variscite.com/product/system-on-module-som/cortex-a53-krait/dart-sd410-qualcomm-snapdragon-410/>, 2019.
- [23]. Pisha L, Hamilton S, Sengupta D, Lee C-H, Vastare KC, Luna S, Zubatiy T, Yalcin C, Grant A, Stambaugh M, Boothroyd A, Chockalingam G, Gupta R, Rao B, and Garudadri H, “A wearable platform for hearing aids research,” in International Hearing Aid Research Conference (IHCON), 2018.
- [24]. Analog Devices, Inc., “ADAU1372 data sheet.” <https://www.analog.com/media/en/technical-documentation/data-sheet/ADAU1372.pdf>, 2014.
- [25]. Bentler R, Mueller HG, and Ricketts TA, Modern hearing aids: verification, outcome measures, and follow-up. Plural Publishing, 2016.
- [26]. Kuk F, Keenan D, and Lau C.-c., “Vent configurations on subjective and objective occlusion effect,” *Journal of the American Academy of Audiology*, vol. 16, no. 9, pp. 747–762, 2005. [PubMed: 16515145]
- [27]. Kemp DT, “Stimulated acoustic emissions from within the human auditory system,” *The Journal of the Acoustical Society of America*, vol. 64, no. 5, pp. 1386–1391, 1978. [PubMed: 744838]
- [28]. Sergi P, Pastorino G, Ravazzani P, Tognola G, and Grandori F, “A hospital based universal neonatal hearing screening programme using click-evoked otoacoustic emissions,” *Scandinavian Audiology*, vol. 30, no. 1, pp. 18–20, 2001.
- [29]. Sonion, “Sonion.” <https://www.sonion.com/>, 2019.
- [30]. Sonion, “VPU14AA01 tentative data sheet.” Private communication, 2018.
- [31]. Lee C-H, Rao BD, and Garudadri H, “Bone-conduction sensor assisted noise estimation for improved speech enhancement,” in *Proc. Interspeech 2018*, pp. 1180–1184, 2018.
- [32]. Kuleshov V, Enam SZ, and Ermon S, “Audio super resolution using neural networks,” arXiv preprint arXiv:1708.00853, 2017.
- [33]. Semiconductor Lattice, “MachXO3: futureproof your control PLD and bridging designs.” <https://www.latticesemi.com/Products/FPGAandCPLD/MachXO3>, 2019.
- [34]. Goldie J, “The many flavors of LVDS (SNLA184),” tech. rep., Texas Instruments, 2011.
- [35]. Peffers M, “Introduction to M-LVDS (TIA/EIA-899),” tech. rep., Texas Instruments, 2002.
- [36]. Huq SB and Goldie J, “Application note 971: an overview of LVDS technology,” tech. rep., National Semiconductor, 7 1998.
- [37]. Lord SE, Weatherall M, and Rochester L, “Community ambulation in older adults: which internal characteristics are important?,” *Archives of physical medicine and rehabilitation*, vol. 91, no. 3, pp. 378–383, 2010. [PubMed: 20298827]
- [38]. Prohaska TR, Anderson LA, Hooker SP, Hughes SL, and Belza B, “Mobility and aging: transference to transportation,” *Journal of Aging Research*, vol. 2011, 2011.
- [39]. Peel NM, Kuys SS, and Klein K, “Gait speed as a measure in geriatric assessment in clinical settings: a systematic review,” *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, vol. 68, no. 1, pp. 39–46, 2013.
- [40]. Meizen-Derr J, Lim LH, Choo DI, Buyniski S, and Wiley S, “Pediatric hearing impairment caregiver experience: impact of duration of hearing loss on parental stress,” *International Journal of Pediatric Otorhinolaryngology*, vol. 72, no. 11, pp. 1693–1703, 2008. [PubMed: 18819718]
- [41]. Bech Christensen C, Hietkamp RK, Harte JM, Lunner T, and Kidmose P, “Toward EEG-assisted hearing aids: objective threshold estimation based on ear-EEG in subjects with sensorineural hearing loss,” *Trends in hearing*, vol. 22, p. 2331216518816203, 2018.
- [42]. Mihajlović V, Grundlehner B, Vullers R, and Penders J, “Wearable, wireless EEG solutions in daily life applications: what are we missing?,” *IEEE Journal of Biomedical and Health Informatics*, vol. 19, pp. 6–21, 1 2015. [PubMed: 25486653]
- [43]. Kondo K, Noonan KM, Freeman M, Ayers C, Morasco BJ, and Kansagara D, “Efficacy of biofeedback for medical conditions: an evidence map,” *Journal of general internal medicine*, pp. 1–11, 2019.

- [44]. Sterman MB and Egner T, "Foundation and practice of neurofeedback for the treatment of epilepsy," *Applied Psychophysiology and Biofeedback*, vol. 31, p. 21, 4 2006. [PubMed: 16614940]
- [45]. Enriquez-Geppert S, Smit D, Pimenta MG, and Arns M, "Neurofeedback as a treatment intervention in adhd: current evidence and practice," *Current psychiatry reports*, vol. 21, no. 6, p. 46, 2019. [PubMed: 31139966]
- [46]. Kim BH, Chun J, and Jo S, "Dynamic motion artifact removal using inertial sensors for mobile BCI," 2015 7th International IEEE/EMBS Conference on Neural Engineering (NER), pp. 37–40, 4 2015.
- [47]. Haykin S, *Communication systems*. John Wiley & Sons, 2008.
- [48]. Analog Devices, Inc., "AD9235 12-Bit, 20/40/65 MSPS 3 V A/D converter." <https://www.analog.com/media/en/technicaldocumentation/data-sheets/AD9235.pdf>, 2012.
- [49]. Kates JM, "Master hearing aid implementation in MATLAB." Private communication, 2016.
- [50]. Kates JM, "Principles of digital dynamic-range compression," *Trends in Amplification*, vol. 9, no. 2, pp. 45–76, 2005. [PubMed: 16012704]
- [51]. Kates JM, *Digital hearing aids*. Plural publishing, 2008.
- [52]. van Waterschoot T and Moonen M, "Fifty years of acoustic feedback control: atate of the art and future challenges," *Proc. IEEE*, vol. 99, no. 2, pp. 288–327, 2011.
- [53]. Kates JM, "Modeling the effects of single-microphone noise-suppression," *Speech Commun*, vol. 90, pp. 15–25, 2017.
- [54]. Greenberg JE and Zurek PM, "Evaluation of an adaptive beamforming method for hearing aids," *J. Acoust. Soc. Am*, vol. 91, no. 3, pp. 1662–1676, 1992. [PubMed: 1564202]
- [55]. Griffiths LJ and Jim CW, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas Propag*, vol. 30, no. 1, pp. 27–34, 1982.
- [56]. Frost OL, "An algorithm for linearly constrained adaptive array processing," *Proc. the IEEE*, vol. 60, no. 8, pp. 926–935, 1972.
- [57]. Greenberg JE, "Modified LMS algorithms for speech processing with an adaptive noise canceller," *IEEE Trans. Speech Audio Process*, vol. 6, no. 4, pp. 338–351, 1998.
- [58]. Hoshuyama O and Sugiyama A, "Robust adaptive beamforming," in *Microphone arrays: signal processing techniques and applications*, pp. 87–109, Springer, 2001.
- [59]. Duttweiler DL, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech Audio Process*, vol. 8, no. 5, pp. 508–518, 2000.
- [60]. Maidment DW, Ali YH, and Ferguson MA, "Applying the COM-B model to assess the usability of smartphone-connected listening devices in adults with hearing loss," *Journal of the American Academy of Audiology*, vol. 30, no. 5, pp. 417–430, 2019. [PubMed: 31044693]
- [61]. Paglialonga A, Tognola G, and Pincirola F, "Apps for hearing science and care," *American Journal of Audiology*, vol. 24, no. 3, pp. 293–298, 2015. [PubMed: 26649533]
- [62]. Charl A and LeRoux B, "Web apps are cheaper to develop and deploy than native apps, but can they match the native user experience?, *communications of the acm*, 54 (5), 49 gartner inc., (2013)," 2011.
- [63]. Lee and Ware B, *Open Source Development with LAMP: Using Linux, Apache, MySQL and PHP*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.
- [64]. Keidser G, Dillon H, Flax M, Ching T, and Brewer S, "The NAL-NL2 prescription procedure," *Audiology research*, vol. 1, no. 1, 2011.
- [65]. Boothroyd A and Mackersie C, "A "Goldilocks" approach to hearing-aid self-fitting: user interactions," *American journal of audiology*, vol. 26, no. 3S, pp. 430–435, 2017. [PubMed: 29049625]
- [66]. Mackersie C, Boothroyd A, and Lithgow A, "A "Goldilocks" approach to hearing aid self-fitting: ear-canal output and speech intelligibility index," *Ear and hearing*, vol. 40, no. 1, pp. 107–115, 2019. [PubMed: 29894379]
- [67]. Dillon H, Keidser G, O'Brien A, and Silberstein H, "Sound quality comparisons of advanced hearing aids," *The hearing journal*, vol. 56, no. 4, pp. 30–32, 2003.

- [68]. A. N. S. Institute, Specification of Hearing Aid Characteristics (ANSI 3.22-2014). ANSI New York, 2014.
- [69]. A. V. 2, <https://www.audioscan.com/verifit2> Audioscan / Etymonic Design Inc., 2014.
- [70]. Knowles Electronics, "RVA-90020-NXX datasheet," 2009.
- [71]. Knowles Electronics, "RVA-90080-NXX datasheet," 2014.

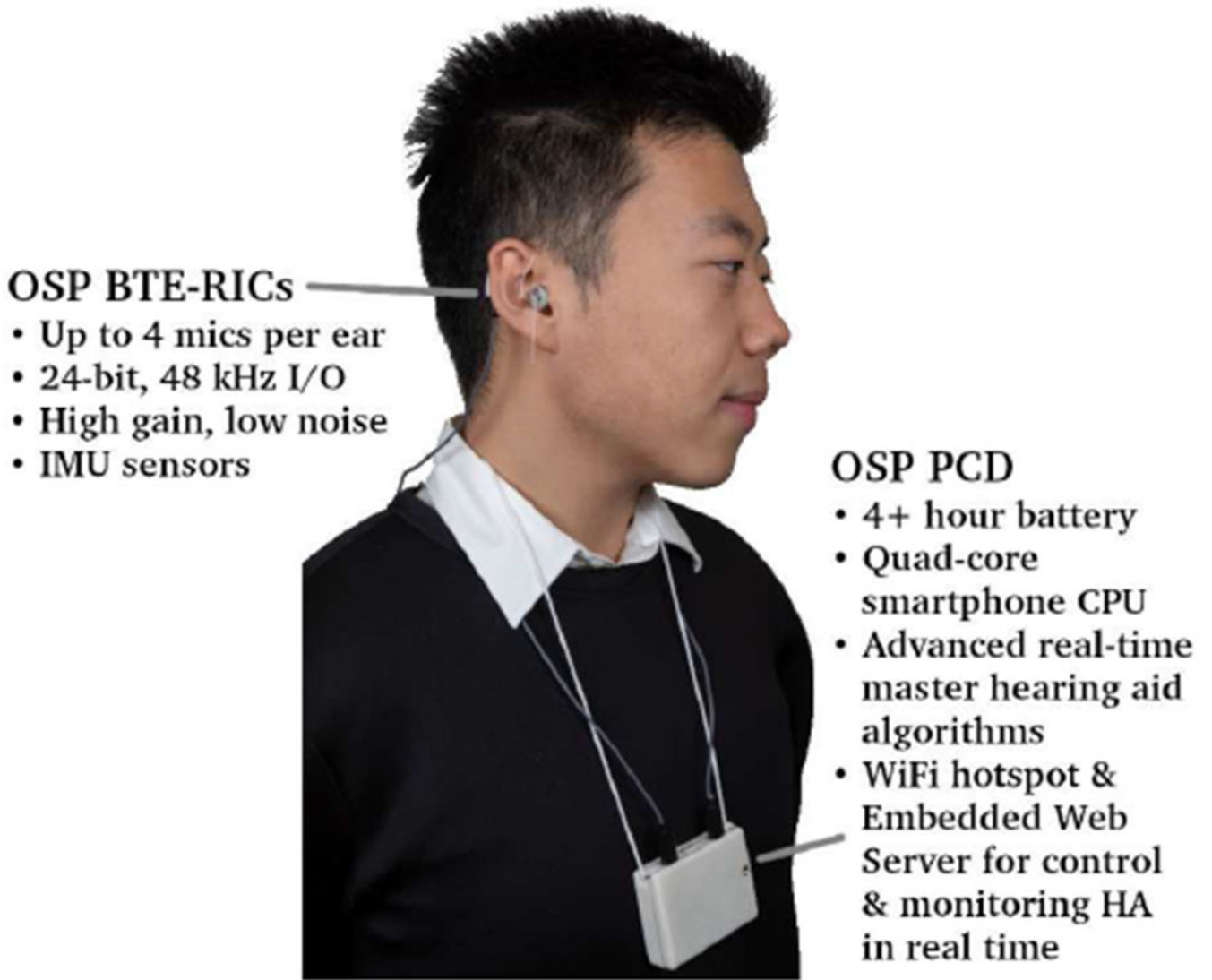


Fig. 1. A user wearing the OSP wearable platform. The two hardware components shown are the behind-the-ear receiver-in-canal (BTE-RIC) transducers and the Processing and Communication Device (PCD).

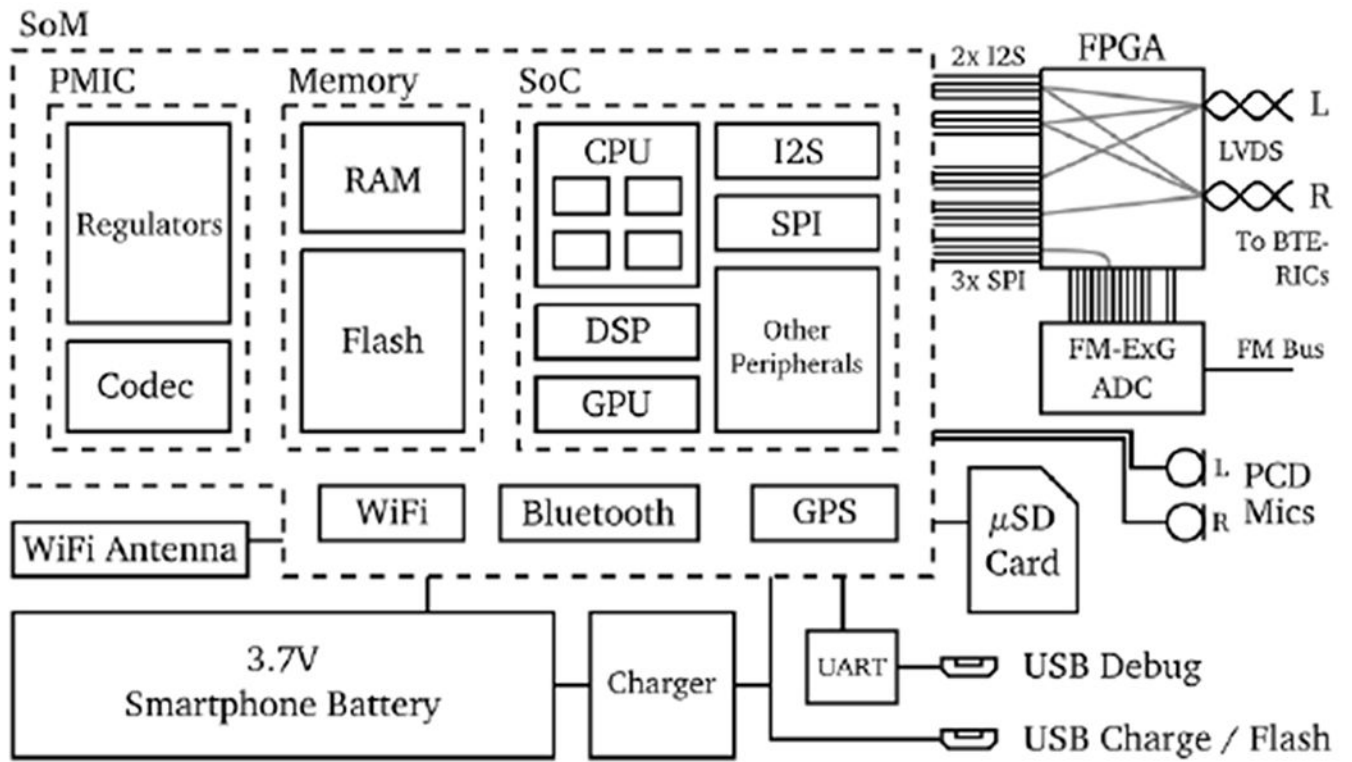


Fig. 2.
Block diagram of the OSP PCD (Processing and Communication Device).

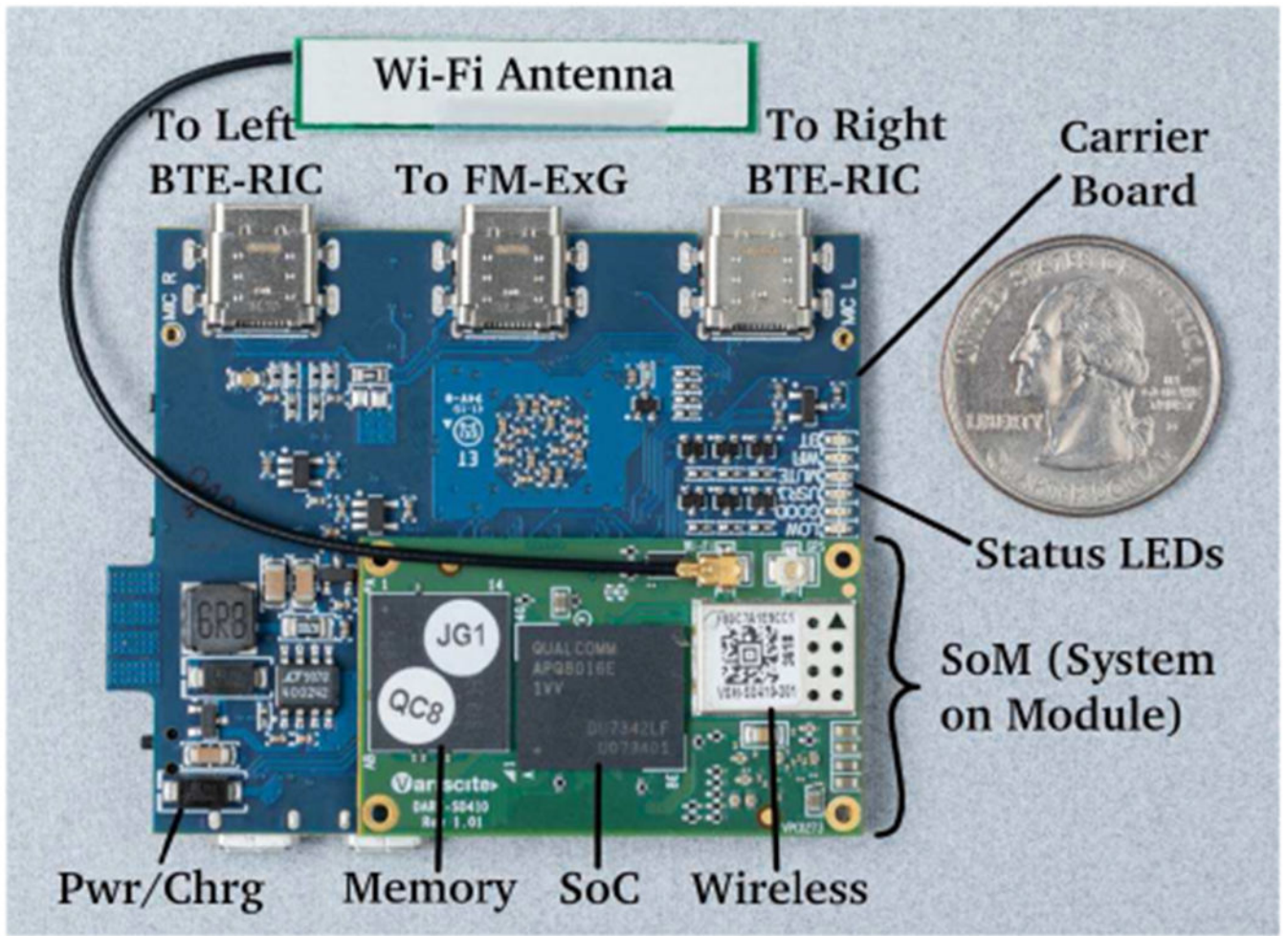


Fig. 3. Components of the OSP PCD (Processing and Communication Device): the carrier board hosting the Snapdragon 410c SoM.



Fig. 4. The OSP PCD disassembled, showing the battery, the back of the carrier board, and the plastic shell.

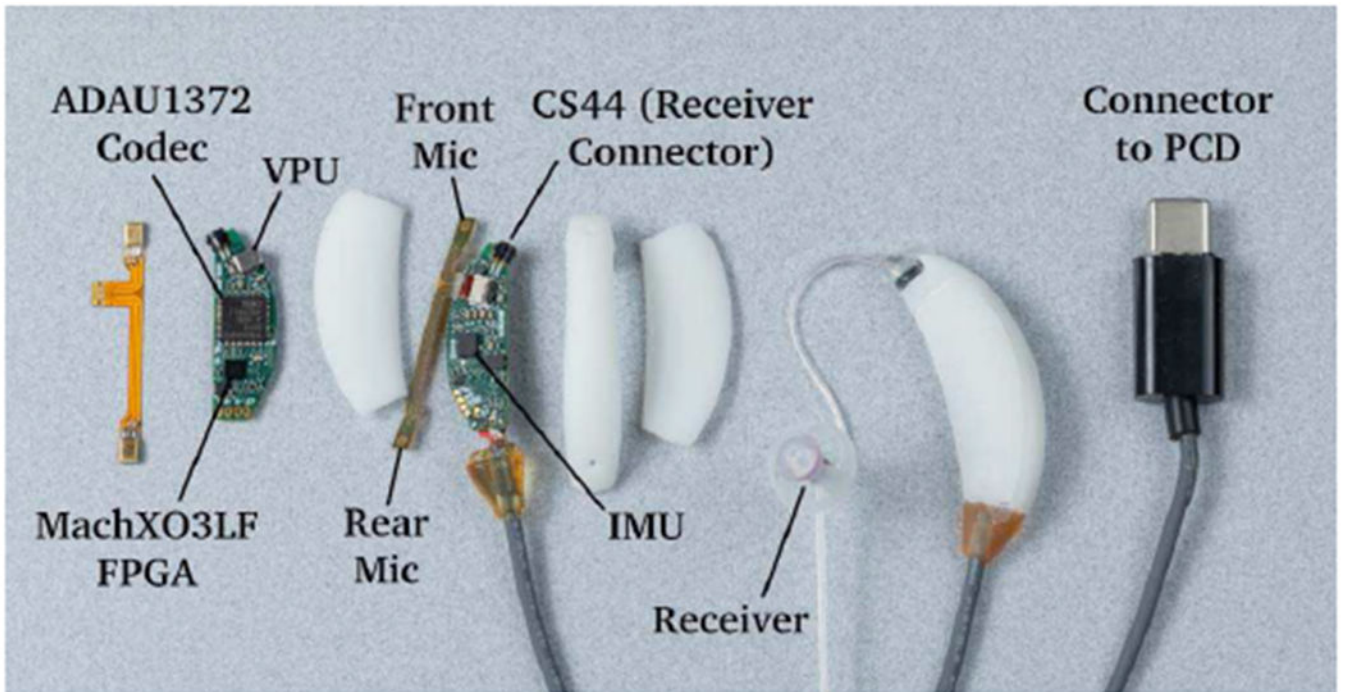


Fig. 5.
The OSP BTE-RICs, together and disassembled.

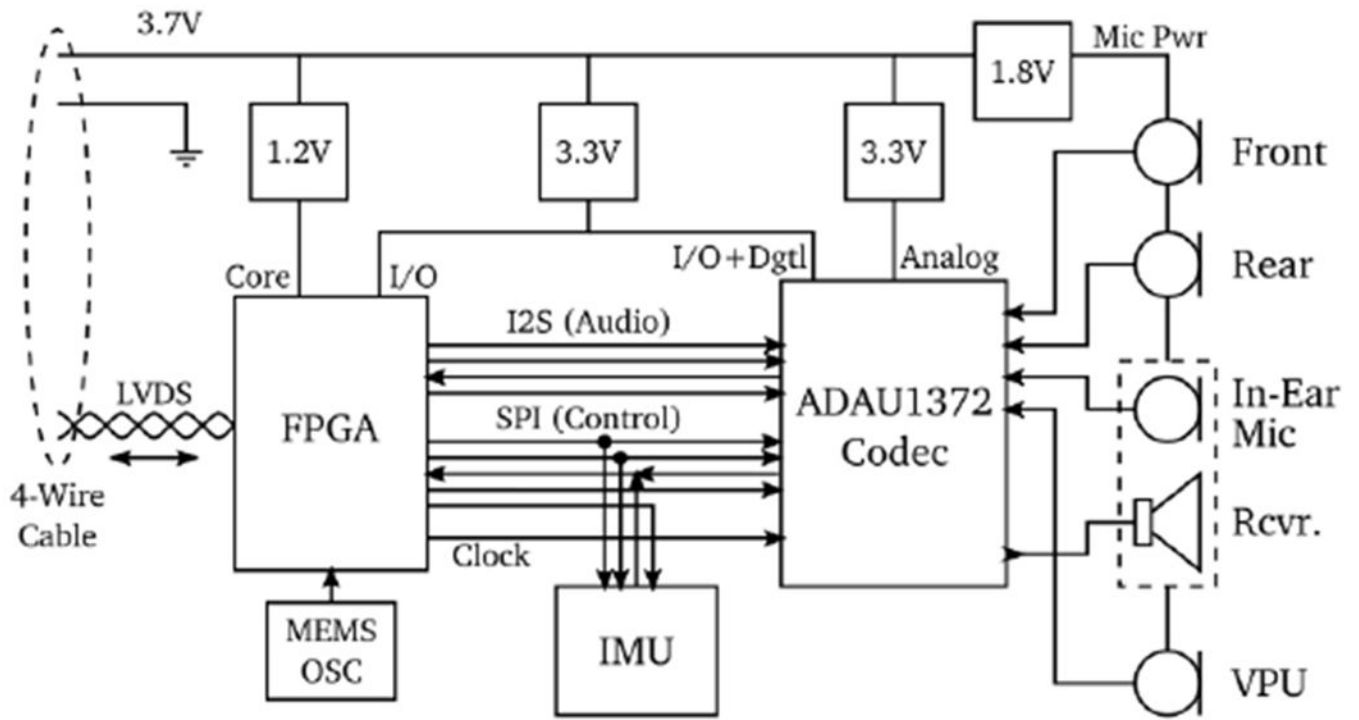


Fig. 6.
Block diagram of the OSP BTE-RICs.

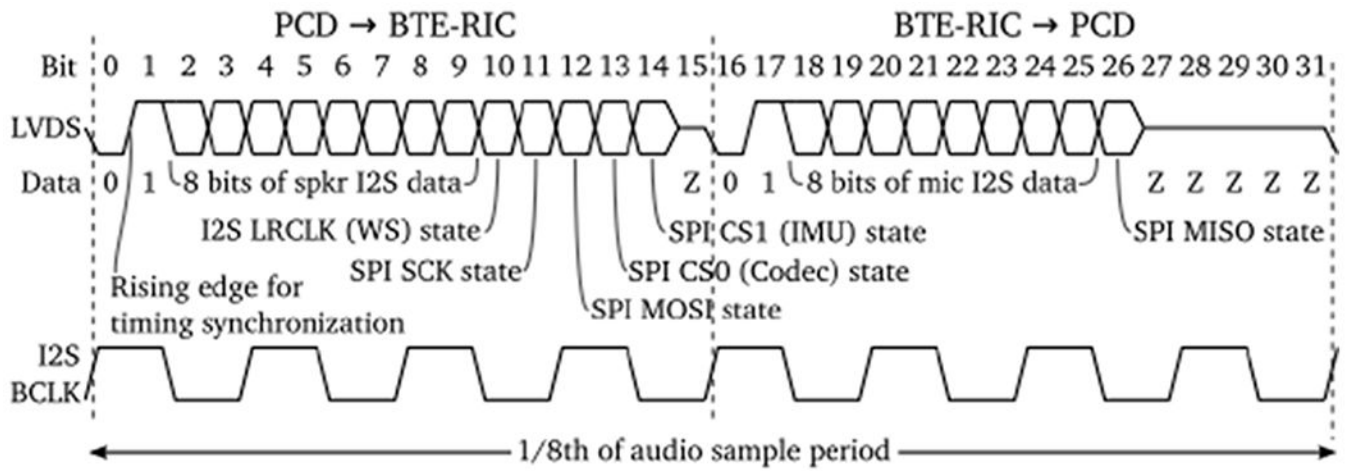


Fig. 7. BLVDS protocol for communication between the PCD and BTE-RICs. 8 bits of I2S audio data are transmitted in each direction, plus a number of control signals, during the same time as 8 I2S bit clocks occur.

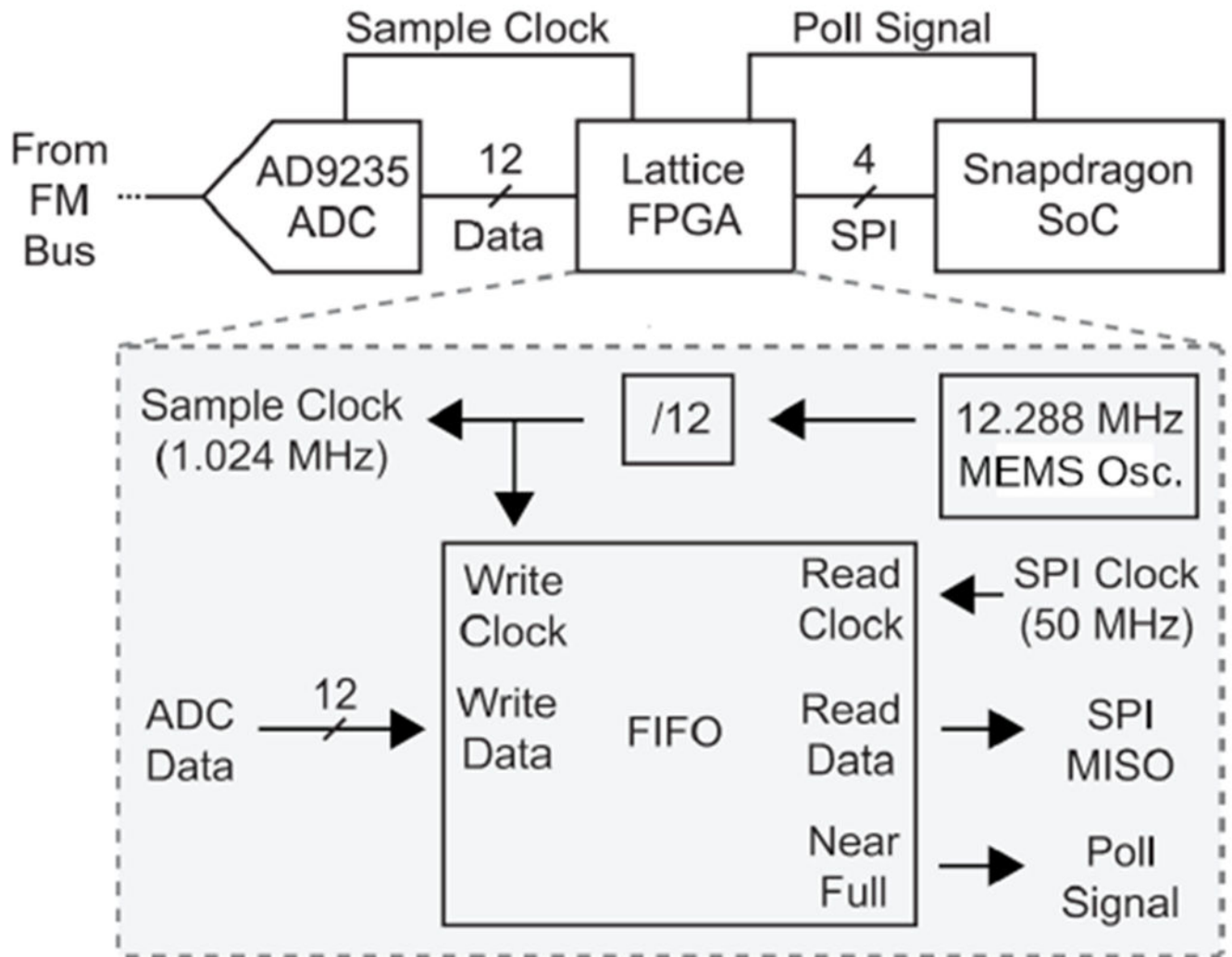


Fig. 8. Block diagram of FM-ExG hardware in the OSP PCD. The FPGA converts between parallel and SPI data formats and stores samples in a FIFO queue for batched access by the SoC. Note that the FM sample clock is derived from the same MEMS oscillator as the I2S audio is, so the ExG and audio streams remain permanently synchronized.

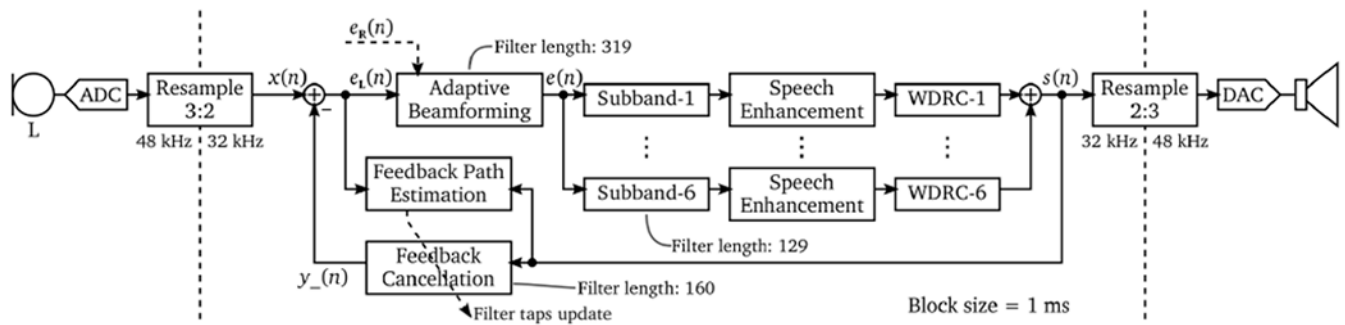


Fig. 9. RT-MHA software block diagram with signal flows. Audio I/O operates at 48 kHz and all HA processing is carried out at 32 kHz. The baseline HA functions provided include adaptive beamforming (BF), subband decomposition, speech enhancement (SE), wide dynamic range compression (WDRC), and adaptive feedback cancellation (AFC). See Fig. 10 for an enlarged picture of the beamforming block.

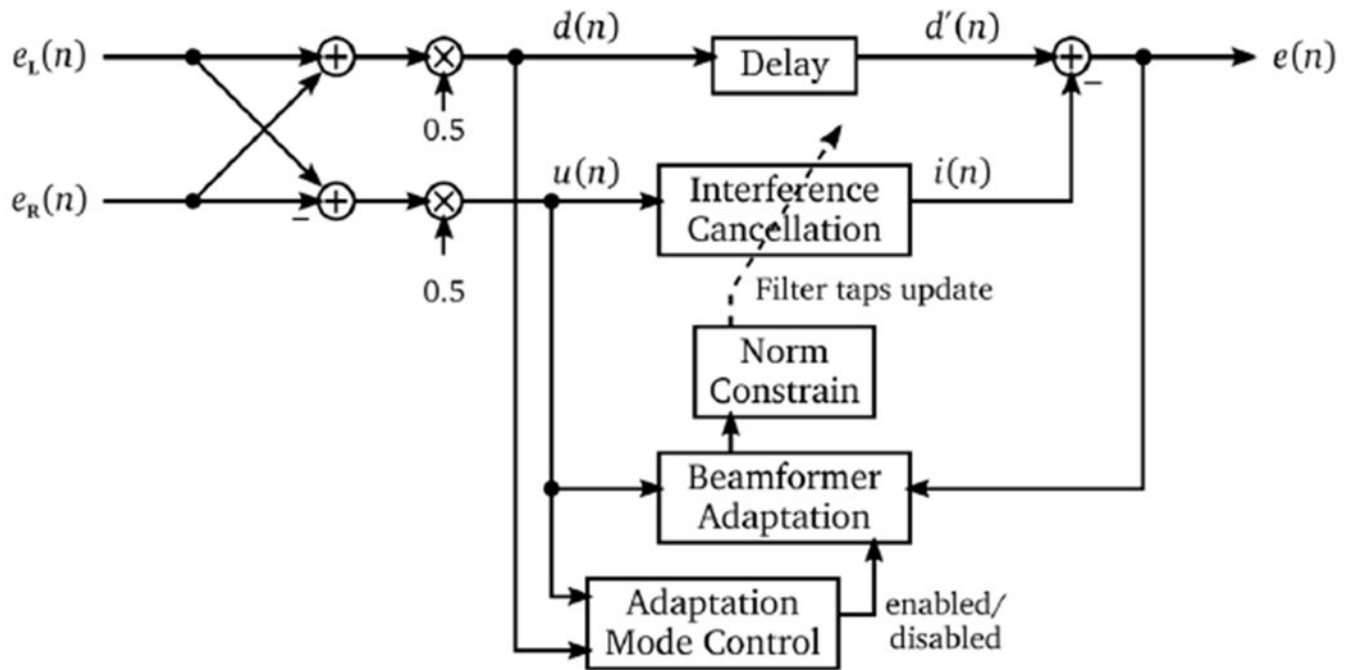


Fig. 10.

The two-microphone adaptive beamforming system in the RT-MHA. Adaptive filtering algorithms are utilized to generate interference estimates based on the left and right channel inputs, which are used to enhance the target signal.

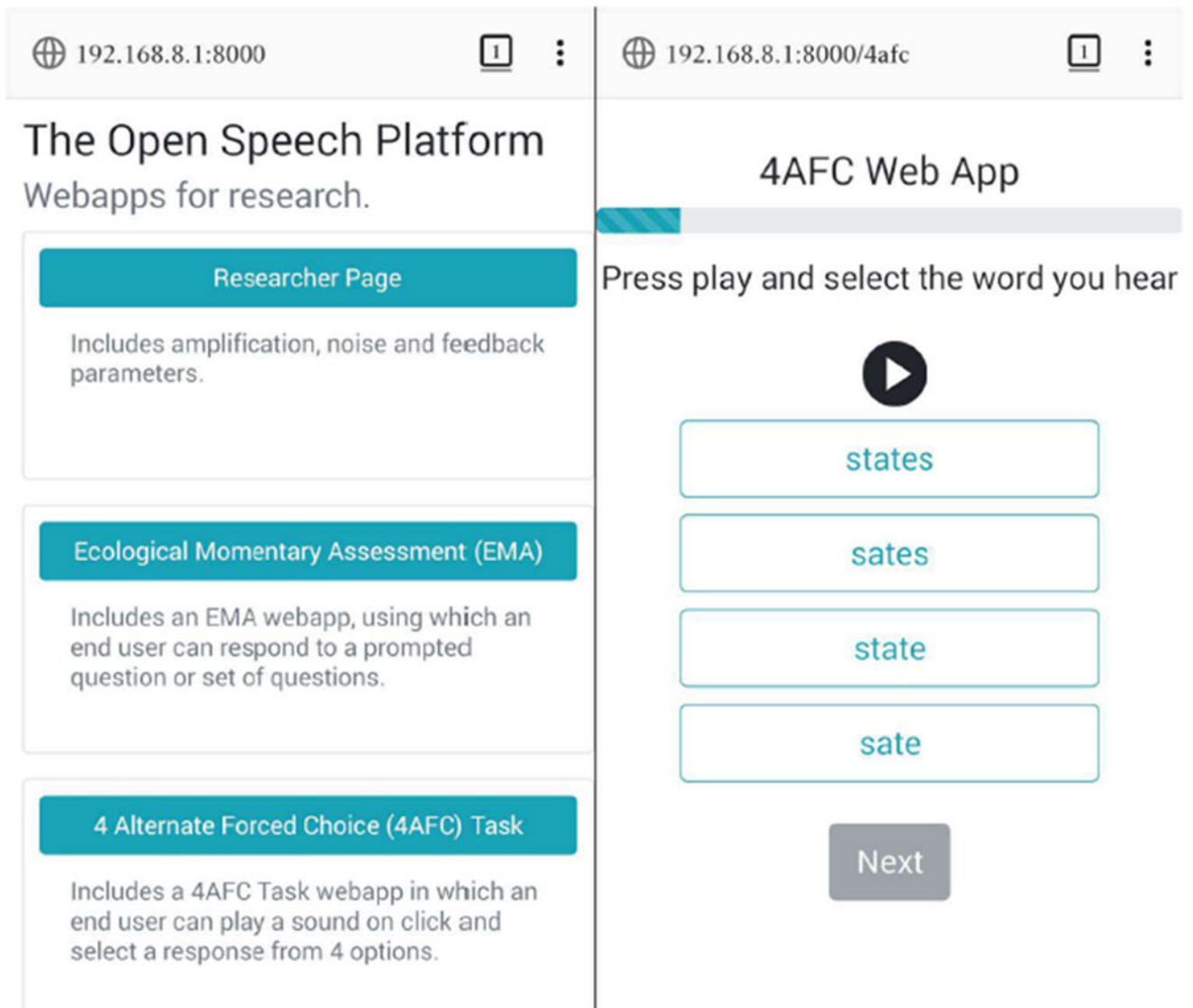


Fig. 11.

Screenshots of the main EWS page and the 4AFC task, taken from a smartphone connected to the WiFi hotspot of an OSP PCD. After powering on the PCD and connecting the smartphone to the new “ospboard-*” WiFi hotspot, the user simply enters “ospboard.local” or “192.168.8.1:8000” into a web browser and receives the page on the left. Clicking on the “4AFC” button and logging in returns the page on the right, which is a fully-functional web app that interfaces with the RT-MHA state in real time.

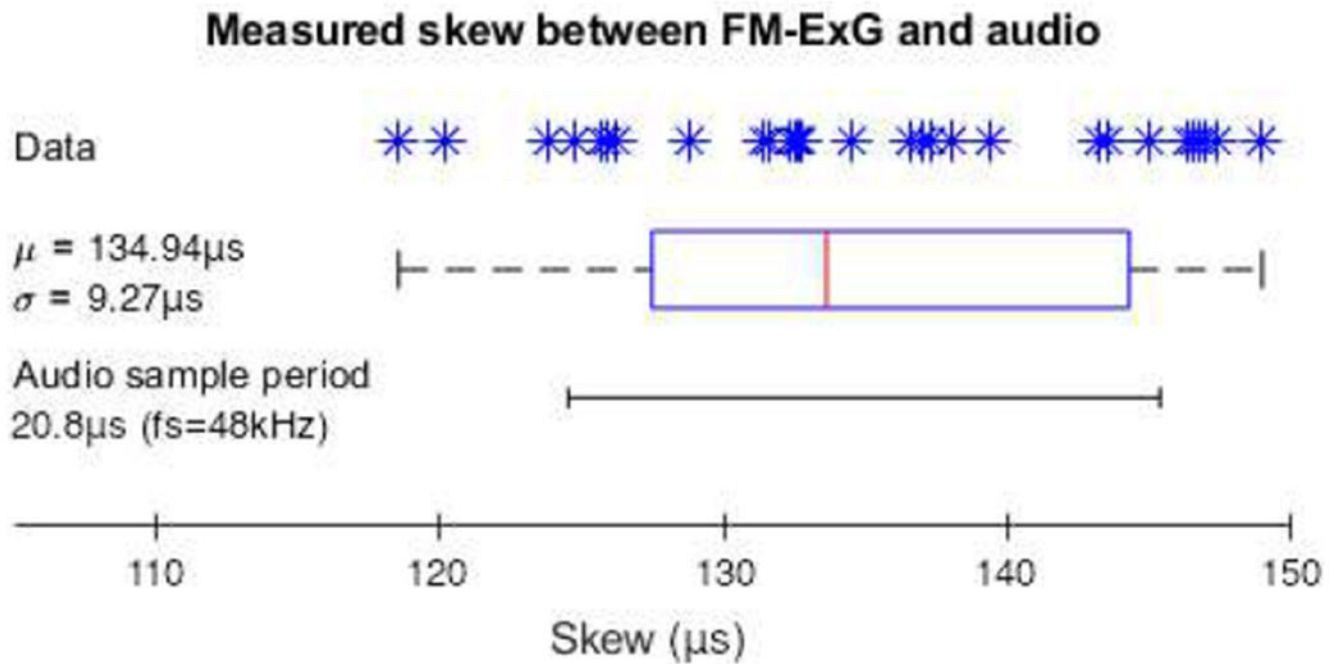


Fig. 12. Comparing 32 trials of the measured skew between OSP's FM-ExG and audio streams with the audio sample period. Since the measurements only vary over about two audio sample periods, OSP can perform simultaneous FM-ExG and audio streaming synchronized to within about 2 audio sample periods, or about $40\mu\text{s}$.

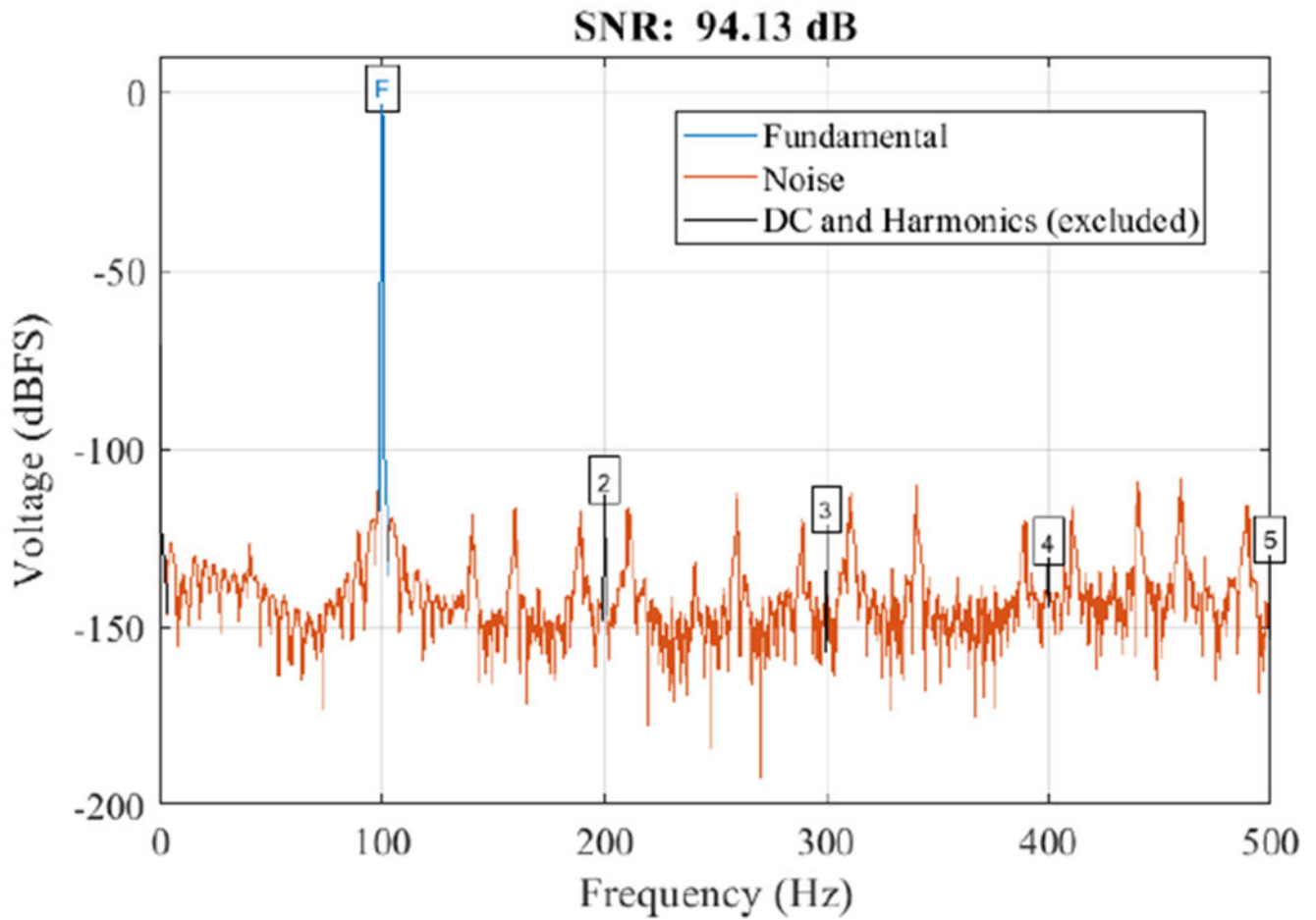


Fig. 13. Example spectrum of demodulated FM-ExG output, for a 100 Hz sinusoidal signal as data. The FM waveform (250 kHz carrier, 20× bandwidth expansion) was generated in software and played into the OSP PCD's analog FM-ExG input via a NI DAQ test device. The sampled signal was recorded on the PCD and demodulated in MATLAB.

TABLE I

ANSI 3.22 TEST RESULTS FOR OSP SYSTEM CONFIGURATIONS AS MEASURED BY AUDIOSCAN VERIFIT 2, AS COMPARED TO RESULTS FROM FOUR COMMERCIAL HAS.

Metric	Units	Commercial HAs				OSP Lab Sys.		OSP Wearable	
		A	B	C	D	X	Y	X	Y
Average Gain	dB	40	40	25	35	40	40	35	38
Max OSPL90	dB SPL	107	112	110	111	121	130	119	129
Average OSPL90	dB SPL	106	109	108	106	112	126	111	125
Average Gain @ 50 dB	dB	37	39	25	35	35	41	34	38
Low Cutoff	kHz	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
High Cutoff	kHz	5	6	5	6.73	8	6.3	8	6.73
Equivalent Input Noise	dB SPL	27	26	30	27	29	28	28	27
Distortion @ 500 Hz	% THD	1	1	0	0	2	1	5	1
Distortion @ 800 Hz	% THD	1	1	0	0	3	2	5	1
Distortion @ 1600 Hz	% THD	0	0	0	0	1	1	2	0

X: with high-bandwidth Knowles receiver [70]

Y: with high-power Knowles receiver [71]

TABLE II

RT-MHA REAL-TIME PROCESSING PERFORMANCE STATISTICS FOR RELEASE 2018C AND THE CURRENT TEST VERSION. WALL-CLOCK TIME TAKEN TO PERFORM EACH PROCESSING STEP ON 1 MS AUDIO BUFFERS.

Operation	Average Time (μ s)		Maximum Time (μ s)	
	2018c	Current	2018c	Current
Downsampling	32	25	91	52
Beamforming	N/A	81	N/A	121
Subband filtering	282	193	822	408
Speech Enhancement	12	12	408	120
Peak detection	30	33	384	228
WDRC	18	14	384	180
AFC	190	142	3661	168
Global MPO	N/A	11	N/A	35
Upsampling	33	25	91	54
Total L or R *	597	536	5841	1366
Overall HA †	694	580	4635	953

* The sum of the above rows; for maximum time, this does not necessarily represent a single real frame, since it is the sum of the maximum values ever recorded for each operation

† The measured total time of the audio processing callback, including sending work to the threads for the left and right channels and waiting for them to complete

TABLE III

CURRENT DRAW (AT NOMINAL 3.7 VDC) AND BATTERY LIFE (ASSUMING 2000 MAH LI-ION BATTERY) FOR COMMON SYSTEM USE CASES.

Test Conditions	Avg. Current (mA) (± 5)		Battery Life (h)	
	2018c	Current	2018c	Current
Idle (WiFi off)	243	220	8.2	9.1
Idle	284	259	7.0	7.7
HA audio loopback	317	289	6.3	6.9
RT-MHA, no BF	440	370	4.5	5.4
RT-MHA + BF	N/A	390	N/A	5.1

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript