

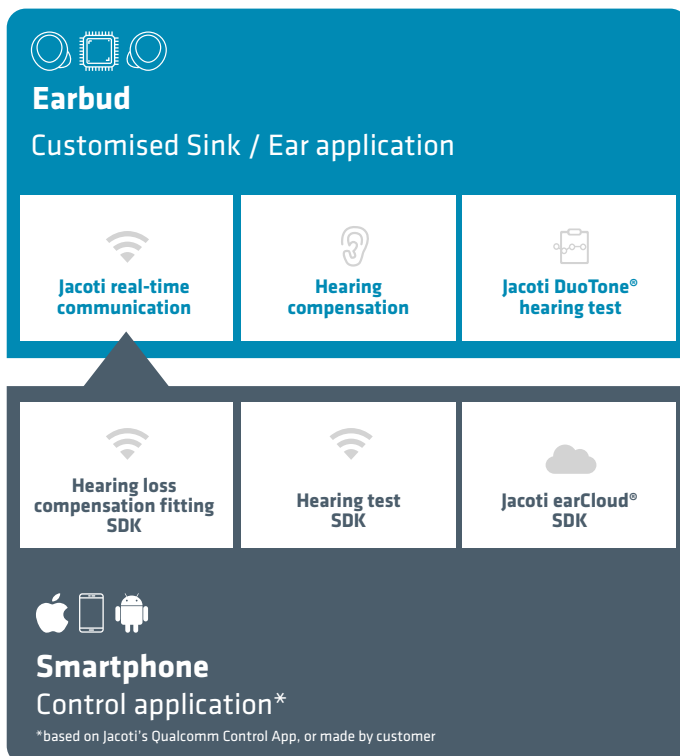


Jacoti Inside

Jacoti hearing personalisation for QCC514x Bluetooth Audio SoCs: True Wireless Headset integration

Overview

- Jacoti brings its hearing technologies to embedded platforms based on QCC514x series.
- Hearing devices such as headsets or hearing aids based on this chip family can perform hearing test, hearing loss compensation and audio personalization.
- Jacoti has ported and optimised its processing engine for the QCC514x series resulting in a low processing and RAM footprint, allowing Jacoti inside to be integrated on-chip in a range of use cases and applications.



Features

Hearing Loss Compensation

Jacoti brings state-of-the-art Hearing Loss Compensation technology to QCC514x headsets. The HLC uses Jacoti HearingKit® audio signal processing technology focused on compensating the individual's hearing profile.

- Compensation optimised for normal hearing to moderate hearing loss: -20 dB to 60 dB
- HLC can be tuned for Hi-Quality or Low-DSP footprint
- Can be used standalone, with the Ambient mode feature
- Ambient mode can be used along Bluetooth streaming (Mixed mode)

DuoTone® hearing test procedure

DuoTone® is a patented pure tone audiometry procedure that employs pairs of pure-tone stimuli at different frequencies

Integration with Jacoti earCloud®

Jacoti earCloud® is Jacoti's Cloud platform that allows to store audiological medical data and is prepared to connect hearing experts with users.

Companion mobile control app

Jacoti Connect+ showcase the interaction of a smartphone remote control app with Jacoti Inside running on-chip.

Headset integration specifications

Integration options	Kymera Capability
Audio Codec	Codec-independent
Configuration	GAIA protocol
Power consumption	Current (mA)
HLC TWS Mirroring (APTx)	6.0
HLC TWS Mirroring (APTx) + Ambient mode	7.5
HLC Ambient mode	4.3
TWS Mirroring (APTx)	5.6
DSP Utilization	MIPS
HLC TWS Mirroring (APTx)	17.8
HLC TWS Mirroring (APTx) + Ambient mode	31.5
HLC Ambient mode	31.4*
HLC HFP	3.8
Hearing Test	22.2
Resource usage	Memory (kiB)
PM usage	35.5
DM, HLC (A2DP)	21
DM, HLC (HFP)	25.1

Notes:

- MIPS are computed using ACAT `profiler.run_clks(30)`
- For MIPS, we show measurements for our integration codebase. These may differ from Qualcomm codebase due to additional modifications required to run Jacoti HLC technology. These represent the worst-case scenario (highest sampling rate - most complex DSP pipeline configuration).
- Hearing test use cases use a separate processing graph that solely involves the QC capability and audio I/O.
- PM and DM are computed using ACAT `heapmem.run_all(), heapmmem.run_all()`
- PM figures include peak consumption at initialisation

Integration into on-board

Hearing Loss compensation can be used as a Kymera capability in all headset use cases: A2DP streaming (TWS Mirroring), HFP voice call along Qualcomm technologies such as Acoustic Echo Cancellation or Clear Voice Capture (cVc) (1, 2 or 3 microphones), and ambient mode.

Our solution runs on a single Kalimba DSP for all use cases.

Jacoti has developed a GAIA-based protocol, allowing smartphone remote control applications to interact with Jacoti technologies running on-chip.

QCC514x Specifications

Bluetooth 5.2

Audio DSP

Dual 120MHz Kalimba audio DSP cores
Flexible clock speed from 2MHz up to 120MHz

Applications subsystem

32-bit firmware processor
32-bit 32/80MHz developer processor

Memory 112KB program RAM, 448KB data RAM

Interfaces

UART, 2x Bit Serializers (I2C/SPI), USB 2.0, SDIO, QSPI, NOR flash, up to 55x PIO

Power Management

Integrated power management unit (PMU)
Dual switch-mode power supply (SMPS)

Jacoti brings medical grade precision to Qualcomm's Bluetooth Audio SoCs, helping to meet increasing consumer demand for hearing enhancement.

Jacoti

HEARING WITHOUT BARRIERS

Jacoti's hearing solutions can be deeply embedded in any consumer electronics device with an audio output. Our technology enhances audio experiences tailored to every customer's individual needs & preferences.

Contact us: business@jacoti.com | press@jacoti.com