Raspberry Pi Zero Deployment Addendum — Hybrid In‑Ear Sensing

This addendum tailors the hybrid in‑ear sensing system (Few‑Shot Siamese + RespEar‑style DSP) for deployment on Raspberry Pi Zero hardware. It covers Pi Zero (BCM2835, 1× ARM1176JZF‑S @1 GHz) and Pi Zero 2 W (BCM2710A1, 4× Cortex‑A53 @1 GHz).

# 1) Hardware Targets & OS

• Pi Zero (v1): 512 MB RAM, single‑core ARMv6, NEON unavailable; use TFLite built for ARMv6 without NEON, or pure‑NumPy/DSP.  
• Pi Zero 2 W: 512 MB RAM, quad‑core ARMv8‑A (Cortex‑A53), NEON available; use TFLite/ONNX Runtime with NEON.  
• OS: Raspberry Pi OS Lite (64‑bit for Zero 2 W; 32‑bit for Zero v1). Disable desktop to reduce overhead.

# 2) Recommended Models & Quantization

• Pathway B (Few‑Shot Reps/Classification):  
 – Primary (Zero 2 W): TinyCNN‑1D INT8 (~0.24 MB) at 100–200 Hz windows; MobileNetV3‑Tiny INT8 (~1.43 MB) if spectrogram‑based and cores permit.  
 – Fallback (Zero v1): TinyCNN‑1D INT8 only; keep kernel sizes small (3–5), channels ≤32, depthwise‑separable where possible.  
• Pathway A (RespEar‑style DSP): fixed‑point/INT16 filters, FFT size 256–512; CPU cost is minimal on both models.

# 3) Expected Latency & Throughput

Assumptions: 16 kHz in‑ear mic, 200 Hz IR/IMU, 1.0 s window, 50% overlap.  
• Zero 2 W (4×A53 @1 GHz, NEON):  
 – DSP RR pipeline: 1–3 ms per window.  
 – TinyCNN‑1D INT8: 4–8 ms per window (single thread); 2–4 ms with 2 threads.  
 – MobileNetV3‑Tiny INT8 (spectrogram): 12–25 ms per window (1 thread); 6–12 ms (2 threads).  
• Zero v1 (1×ARMv6 @1 GHz, no NEON):  
 – DSP RR pipeline: 3–6 ms per window.  
 – TinyCNN‑1D INT8: 15–35 ms per window. Use 1.0 s windows with 0.5 s hop to stay real‑time.

# 4) Power & Thermal Estimates

• Zero 2 W: 280–350 mW idle (Lite OS, Wi‑Fi idle); 450–700 mW under ML bursts with 1–2 cores active.  
• Zero v1: 160–220 mW idle; 300–450 mW under load.  
• Budget for audio front‑end + IR/IMU + BLE: add 40–120 mW depending on peripherals and duty cycle.

# 5) Audio & Sensor I/O Settings

• In‑ear mic: 16 kHz mono, 16‑bit PCM. Frame 20–40 ms; STFT 25 ms window, 10 ms hop; 64–80 Mel bins if spectrogram model is used.  
• IR/PPG: 200–400 Hz; decimate to 100–200 Hz for ML; apply IR notch if needed; per‑session DC detrend.  
• IMU (optional): 100–200 Hz; align timestamps using clock sync; store as int16 to reduce memory.

# 6) Memory Budget

• Model (TinyCNN‑1D INT8): ~0.24 MB  
• Audio ring buffers (16 kHz × 2 s × 2 bytes): ~64 KB  
• IR/IMU buffers (200 Hz × 2 s × 2 bytes × 2–3 chans): 160–240 KB  
• STFT/Mel (if used): 200–400 KB working  
• Code + OS headroom: >100 MB available; total footprint well within 512 MB RAM.

# 7) Software Stack

• Inference: TensorFlow Lite (Zero 2 W with NEON); for Zero v1, build TFLite without NEON or use ONNX Runtime Mobile if supported.  
• DSP: NumPy/SciPy or kissfft/fftw; prefer fixed‑point where possible.  
• Runtime: Python for prototyping; migrate hot paths to C++ or Cython for Zero v1.  
• Scheduling: Use separate threads/processes for DSP and ML; pin ML to a dedicated core on Zero 2 W.

# 8) Telemetry & Edge‑to‑Cloud

• BLE for on‑body earbud link; Wi‑Fi for upload when available. Batch metrics (reps, RR, cadence) every 5–10 s; avoid raw audio upload.  
• Optional LLM API in the cloud for coaching summaries; keep on‑device a compact rules engine for offline feedback.

# 9) Validation Targets

• Reps: ≤±1 rep error per set; F1 ≥0.9 for peak detection on seen exercises; ≥0.8 on unseen few‑shot tasks.  
• Respiration: MAE ≤2 BPM sedentary, ≤3 BPM active; Bland–Altman limits within ±6 BPM.  
• Runtime: Zero 2 W end‑to‑end <20 ms per 1 s window; Zero v1 <50 ms per 1 s window.