Open Source Audio Processing Challenge
A General Purpose Pipeline for Interfacing the Tympan with an External Computer

Danielle Benesch, Kocherla Nithin Raj
CRITIAS (critias.ca)

Motivation

Certain applications, such as sound event detection, binaural sound localisation, etc, require the use of algorithms that are resource-heavy (e.g. Machine Learning/Deep Learning algorithms), and thus, can’t run on the Tympan directly. These algorithms can, however, run on an external computer, and the decision made by the algorithm can be relayed back to the Tympan where necessary changes can be made in the audio processing blocks.

We want to create such a pipeline, where input audio can be transmitted to the external computer from the Tympan and then the output of the algorithm can be transmitted back to the Tympan as simple commands, thus alleviating the Tympan of any resource-heavy processing such as detection, estimation, etc, and making it focus only on audio processing.

By using an external computer in the pipeline, the audio processing latency would be independent from the algorithmic latency (detection, estimation etc.). Hence, our target applications would be those which require low audio processing latency but are not as sensitive to algorithmic delays.

Proposed Approach

We plan to use the Tympan to acquire input audio data from the on-board and/or external microphones with the help of the I2S protocol abstraction provided by the Tympan library. Functionality can be provided to vary the number of input channels (1 – 4). This input data will be transmitted to the external computer algorithm, which will be in either MATLAB or Python, via USB/Serial communication. An example algorithm could be alarm sound detection. Using the transmitted audio, the alarm sound would be detected, determining the audio processing parameters which are then sent back to the Tympan as commands where an audio processing algorithm built using the MyAudioAlgorithm template amplifies the alarm sound (or other vital sounds) based only on the command(s) received.

As shown in the Fig. 1, the audio flow from input to output is decoupled from the external algorithm, which merely sends commands to tweak the audio processing parameters on the Tympan. A possible extension of the project could include using the bluetooth protocol for wireless bidirectional data transmission and then comparing latencies across protocols.

Desired Result

We hope to be able to demonstrate the described pipeline with an illustrative example at the ASA conference. We plan to share our software via a public repository on CRITIAS’ GitLab as well as Tympan’s GitHub.