A.3. Insertion of Pseudorandom Noise Values

When, for a transform coded block, the single-bit noise parameter flag does not indicate usage of TNS, an additional three bit are read from the bitstream, indicating the noise level (in $1/8^{th}$ of the step-size signalled for the block) by which the inserted noise is to be scaled. The spectral start index of the noise insertion process equals $1/8^{th}$ of the spectral length, as with TNS filtering, and the pseudorandom number generation starts at the highest spectral index (length – 1) and proceeds downward up to, and including (if applicable), the spectral start index. Within that spectral range, a pseudorandom value is only generated, and the generator state updated, for those transform coefficients quantized to zero.

The pseudorandom number generator is implemented using the simplest, and computationally least complex, realization of a linear congruential generator (LCG; for details see, e.g., P. L'Ecuyer, "Tables of Linear Congruential Generators of Different Sizes and Good Lattice Structure," *Math. Comp.*, vol. 68, no. 225, pp. 249–260, Jan. 1999). In each block, a 32-bit unsigned LCG *state* variable is initialized to (*blockOffset* >> 3) | 1, where *blockOffset* is the block's time offset as a sample index into the given Intra-coding (i.e., random-access) interval being decoded, and | denotes the binary OR operator. Then, a transform sample vector of zeros and length equal to the spectral (i.e., block) length is created, and for every index i in the abovementioned spectral range at which the block's quantized transform value equals zero, *state* is updated to $state \cdot 851723965$ and the 16 LSBs of the resulting value are stored, as a signed integer, at i in the transform sample vector. Next, all generated values in the vector are scaled by $min(2^{16} - 1, \text{noiselevel} \cdot 2^{-16})$, and the same inverse trigonometric transform as is applied to the (de) quantized transform coefficients is invoked on the vector, resulting in an unfiltered noise time-signal of length equal to the block length. For all other blocks where the noise insertion is not applicable or active (e.g., because TNS is used), the noise signal is set to zero as no noise parameters are signalled.

The block-wise noise time-signals are, individually for each channel, FIR filtered by means of the signalled noise filter coefficients, which are encoded, decoded, and clipped exactly like the TNS filter parameters. The filtering process continues across all block boundaries, i.e., a FIR zero input response (ZIR) signal is calculated at the end of each block and added to the filter output at the start of the next block in the given channel (unless the next block begins a new Intra-coding (random-access) interval).

As final step, the resulting filtered noise signal for each channel is added to the output waveform to be written to the output file or pipe, clipping the result to the value range of the PCM input/output bit-depth before the write-out. Note that no noise is added to the decoder's internal prediction buffers (i.e., the decoded waveform buffers) in order to avoid degrading the codec's prediction performance. Note that, while the noise insertion process causes the results of measurements with objective metrics to drop, it does serve to increase the perceptual fidelity of the decoded waveforms in comparison with the **uncompressed** original waveforms. Hence, upon visual inspection by experts (e.g., clinicians), it is believed to be much more likely that those experts do not see a difference between the original and compressed waveforms when noise insertion is being applied, thereby allowing to lower the bit-rate.

A.4. Supplementary FLAC Files for Demonstration

The visual demonstration of the perceptually optimized CfP response on page 1 of this document (underneath the abstract) is accompanied by longer versions of the depicted waveforms, to allow for external comparisons in software viewers/editors. These waveforms, containing $96 \cdot 1024$ samples per channel, were rate matched during encoding and stored losslessly as FLAC files [3] for compatibility.

A.5. Averaged Rate-Distortion Graphs per Dataset

The remaining nine pages of this document depict rate-distortion graphs, with the rate specified in bits per sample and the distortion calculated according to the three metrics defined in [1]. The first 3 graphs (MITECG) are for the 2-channel ECG, the next 3 graphs (OzRaw) for the 4-channel EMG set.