Chris Jiang 400322193
Sohel Saini 400327615

fmMonoBasic

The script was modified to improve signal processing efficiency and adaptability. The sampling frequency was set to 240 kHz, with a cutoff frequency of 16 kHz and a filter order of 101. A decimation factor of 5 was applied to optimize data processing. A custom low-pass filter was implemented using custom_coeff() and custom_low_filter(), replacing the default method to provide greater control over filtering. The input file was updated to "samples4.raw", and the il_vs_th flag was set to 1 to enable take-home mode. Additional modifications were made to enhance code readability, maintainability, and overall signal processing accuracy.

fmMonoBlock

The updated code improves FM demodulation by replacing prev_phase with prev_I and prev_Q, making it more efficient by avoiding the arctan function. A new FIR filter with a hamming window is added for better mono audio extraction and proper downsampling and state-saving (state_I and state_Q) to maintain continuity between blocks. Additional PSD plots now visualize the demodulated FM signal, filtered mono audio, and final downsampled output, providing better insight into the signal processing steps. The updates ensure consistent precision (float32 or float64) between Python and C++ to prevent compatibility issues.

estimatePSD

In our version of the code, we implemented the impulseResponseLPF and convolveFIR functions in filter.cpp, providing the logic for calculating the impulse response and performing convolution, respectively, which were left as placeholders in the original. We also added the computePSD function in fourier.h and its corresponding implementation in fourier.cpp to compute the Power Spectral Density (PSD) by first calculating the Discrete Fourier Transform (DFT). In experiment.cpp, we integrated the computePSD function to compute and log the PSD, enhancing the original code to also track spectral density along with time-domain and frequency-domain data. These changes improved the signal processing capabilities of the code.