Status Reports

COMBSeq - C++ JUCE project

Status Updates

27th November 2024

Done:

- set-up JUCE environment with Visual Studio; compile targets: VST3 & Standalone
- implemented simple white noise generator
- set-up state management via APVTS, tested with volume control for noise generator

Goals for next weeks:

- implement delay processor
- find references for implementing a LFO / Sequencer
- research Serum's Comb filter implementation (How do they derive their visualization? Are they using a different approach from a delay line?)

10th December 2024

Done:

- Implemented sine pulse generator for testing purposes
- Implemented simple delay processor (no measures against discontinuities implemented yet)

Thoughts:

- Delay processor still causing audio artifacts after ~2sec, needs more debugging
- Did not expect to write CSS-like code in C++ for UI Design
- Thinking in samples / buffers for DSP-programming needs more practice to get into the workflow

21st January 2025

Done:

- Wrote stereo-wrapper class for delay processor
- Implemented 4-point interpolation for subsample reading
- Implemented parameter-smoothing

Thoughts:

- Clicks noticeable during live-parameter changes when tested with sine-pulse generator
- Weird stereo behaviour by delay processor

Status Reports

COMBSeq - C++ JUCE project

4th February 2025

Done:

- Implemented simple comb filter processor by adding original and delayed signal
- Live frequency changes now without clicks due to fixed delay processor

Thoughts:

- Planned sequencer-functionality may go out of scope due to time constraints

Final Retrospective

Learned:

- Thinking in Audio-Buffers for realtime programming
- C++ class structure for Processors and Editors, compartmentalization in JUCE
- Fundamental concepts of multithreading, keeping the audio thread lock-free, avoiding priority inversion
- Implementing a realtime delay line, applying it to a comb filter processor

Project goals:

• Realtime Comb-Filter with parameter smoothing	✓
• Automation via internal sequencer	X
• Usable UI	X

Final Thoughts:

- Implementing a usable delay line took a lot more time than anticipated
- Scared of doing UI-design with CSS-style C++
- Debugging DSP is painful