Design of Digital Music Synthesizers

Applying digital signal processing techniques to generate new and interesting musical sounds.

Description

The intention of this creative work is to bridge the gap in learning between my Electrical Engineering, Computer Science, and music knowledge by creating two electronic instruments: a software synthesizer plugin that can be easily integrated with digital audio software, as well as a physical digitally-driven synthesizer that could be used in live performance situations. Additionally, I will share my new knowledge through a guide that others can use to assemble the device & software themselves. In the end, I expect to have an actual set of devices to show for it and newfound knowledge of digital signal processing techniques, music theory, and other concepts in the realm.

The products and the guide would be primarily geared towards people who have both musician and engineer backgrounds, that may also be interested in creating their own unique device for learning and/or performing with.

I was prepared for this project mostly by my studying of the topics in my own time. I stopped music lessons at the age of 13 but continued playing, teaching myself new instruments and techniques along the way. The same goes to say for much of my Computer Science and programming knowledge, which I started teaching myself during my third year of high school and continued up to now. The last component I believed I needed was the Electrical Engineering knowledge, which I have been exposed to within the last year and a half of Electrical Engineering courses here at SMU.

The final product will comprise of three parts: a software synthesizer plugin, a digital music synthesizer, and a guide of the knowledge attained. The former would be used as a starting block to establish the ideas that will be incorporated in both synthesizers, including oscillators, filters, and other components. Its input will be either a standard MIDI keyboard or another non-music control surface (joystick, gamepad, etc). The second will mimic the sound and components of the first, but assembled into a physical standalone contraption, with a homemade control surface. Lastly, both will be accompanied by a guide (most likely an interactive website or blog) that will demonstrate the concepts and implementation details of each component and the overall integration of both synthesizers through examples, graphs, code snippets, and images.

My chosen mentor, Scott Douglas, and I discussed a meeting schedule and decided to meet bi-weekly on Tuesday afternoons to discuss the current progress of the project and what the next steps should be.

In the end, I hope to attain new or furthered knowledge in the following subjects: digital signal processing, Linux/Windows audio stack or VST plugin development, MIDI slave/instrument programming, electronic debugging equipment (ex: oscilloscopes), 3D printing, and circuit design.

Methodology

- Phase One - Software
    - Decide on a platform - Windows/Linux, Plugin/Standalone
    - Research said platform and how to program for it
        - How to provide sound to the host, how to receive MIDI input, etc.
    - Research components of a synthesizer and decide which to incorporate
        - This will define the sound, customizability, difficulty, and breadth of the project
    - Implement each component a step at a time in code, documenting the changes in sound at each step
- Phase Two - Hardware
    - Research ways to implement each individual component
    - Implement the components one at a time, testing them in isolation from the rest
    - Assemble the entire circuit to confirm it works as a whole
    - Create a housing and control surface (keyboard, theremin-like interface, etc) to house the device (using wood/3D-printing/other materials)
    - Assemble the circuit inside the housing and test
    - (Optional) create a MIDI interface to control the device using a host computer
- Phase Three – Website/blog
    - Isolate each implemented component and acquire graphs (time & frequency) and sound samples of the sounds they produce
    - Write a description about the purpose of each component
    - Provide diagrams or code snippets on how each was implemented
    - Compare the software vs the hardware qualities, subjectively and objectively
    - When completed, post the website publicly for others to peruse
- Extra - Analog Components
    - If I have time to spare, it would be an interesting bonus to implement some of the synthesizer components using analog components and to compare the qualities of analog vs digital in regards to their sound characteristics

Timeline

- February 15, 2016 - Proposal turn in, start work
- May 15, 2016 - Software check in
- August 15, 2016 - Software synthesizer complete
- December 1, 2016 - Hardware check in
- January 15, 2017 - Hardware synthesizer complete
- February 1, 2017 - Website check in
- February 15, 2017 - Guide website complete