Mel Murphy
David Hanson
ECE 544
Prof. Kravitz and Prof. Burnett

Final Project Proposal:
Expanded MIDI FPGA Synthesizer

Project Description

Our goal for this project is to build a synthesizer on the Artix-7 FPGA. The synthesizer will accept MIDI signals on an interrupt basis and write them to a sequence. It will then play notes out of the PMOD I2S jack corresponding to the MIDI signals it stored. It will also have MIDI output functionality such that it can send MIDI signals to external instruments. The buttons, switches, and rotary encoder will be used to set parameters of the sequence, including tempo, and these parameters will be displayed on the seven-segment display and the OLED RGB display. The software will be coded in C, and it will run on a Nexys A7-100T FPGA board. We'll use Vivado to build the hardware, and Vitis to build the software.

Design Approach

To build the synthesizer, we'll need four core pieces: MIDI input/output hardware, the processing and timing systems, the signal generator, and the audio output hardware. The MIDI input hardware can be easily adapted from an open-source MIDI receiver module created by Andy West. We'll need to add interrupt functionality to this IP and then package it for use in Vivado. MIDI output is easily achieved using a UART. The processing and timing systems will consist of preexisting Vivado IP -- FIT and AXI timers, a Microblaze processor, an interrupt controller, and associated hardware. The signal generator can consist of either hardware or software, and this implementation will depend on the layout of the audio output hardware, which will be the PMOD I2S jack controlled by a Vivado IP block.

Stretch goals include modulation and visualization capability. The DSP blocks may make it possible to include filter-based effects such as a phaser or flanger, and by sampling and mixing audio signals it may be possible to create delay and reverb effects. We may even be able to modulate these effects using the onboard accelerometer. A series of bandpass filters modulated by the onboard microphone could function as a vocoder. Visualization of the waveform may be achieved using the vivado FFT IP, and that or other types of artistic visualization using the audio signal or tempo could be presented on the OLED rgb screen or a VGA display. Which stretch goals we aim for will likely depend on the difficulty of using the DSP-related IP.

To demonstrate success, we'll show that the sequencer can read input signals from an instrument through MIDI, play a corresponding sequence over the I2S audio, and also play a corresponding sequence to an external instrument using MIDI. If we start running out of time, we'll focus only on our base goals, ensuring that we have functional sequencing, sound generation, and MIDI output. With the preexisting hardware and codebase from Mel's ECE 540 RVfpga project in the Winter, we should be able to accomplish this at the minimum.

Milestones

To stay on top of the project deadline, we'll try to achieve the following milestones:

Week 8 (5/16 - 5/22):

- Submit project proposal
- Package the MIDI input hardware as an AXI module for use in Vivado
- Design basic processing and timing functionality so that the sequencer codebase can be adapted for use with a Microblaze

Week 9 (5/23 - 5/29):

- Design sound generator
- Add I2S output functionality
- Add MIDI output functionality
- Add button, switch, LED, OLED screen, and SSD functionality

Week 10 (5/30 - 6/5):

- Pursue stretch goals
- Fine-tune sequencer and sound generator

Finals Week (6/6 - 6/9):

- Prepare demo
- Prepare project report