FMPGA: The Frequency-Modulating Programmable Gate Array

Joe Finn, Eric Schneider, Manav Trivedi

Problem

- Hardware Synthesizers:
  - Much more limited in scope since they can't be changed, but are portable
  - No way of visualizing sounds
- Software Synthesizers:
  - Can do a lot more, but require you to lug around a computer
  - Requires an expensive Digital Audio Workstation as a prerequisite
- Both are expensive ($250-$1500)

Arturia Minibrute 2,
a $450 entry-level hardware synthesizer

Ableton Live, a DAW that can run a software synthesizer. Costs $400-$800

Our Solution

ECE Areas: Hardware, Signals (Software for verification)

- Use an FPGA to perform digital synthesis using MIDI input
- Offers subtractive synthesis, frequency modulation, filtering, and more
- All inclusive, portable system (unlike software)
- Cheap and modular (unlike traditional hardware solutions)
- Has a built in screen to display waveforms & audio effects (unlike hardware)

Requirements

Prototype

- < 10ms latency between MIDI keyboard press and audio output
- < 1% deviation in frequency from equal temperament tuning
- Achieve 48KHz 16-bit audio output
- OLED and knobs to display and adjust settings
- 8 wavetable oscillators (sawtooth, sine, triangle, square, noise)

Final

- 4-note polyphony
- Apply digital low pass filter, high pass filter, and distortion to an audio stream
- Modulate pitch, amplitude, and supported audio effects
- 12 envelopes, 3 low frequency oscillators to use as modulation sources
- Save and load at least 10 presets in RAM

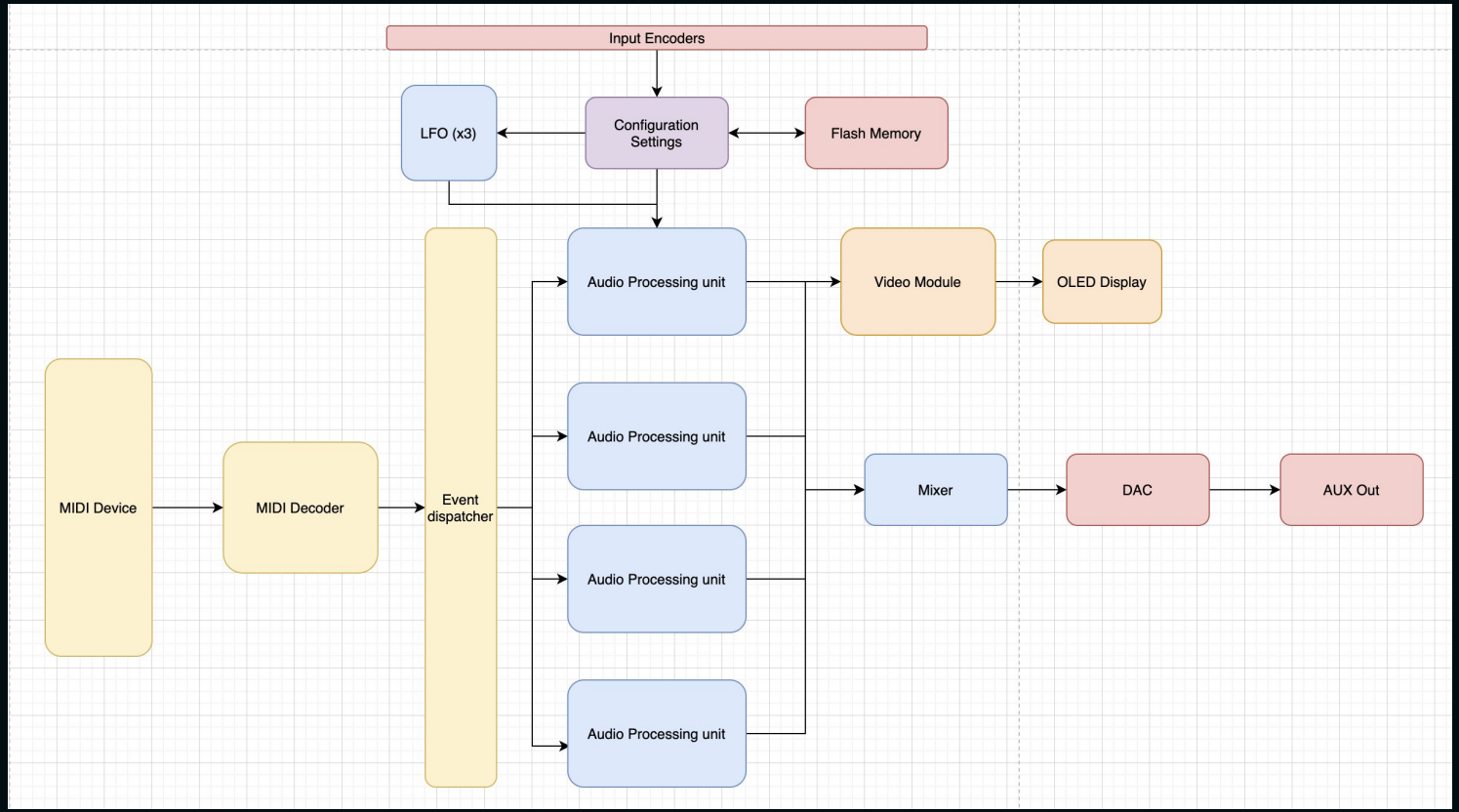
Technical Challenges

- Efficient use of on-board multipliers
- Handshaking between many complex modules
- Interfacing with FPGA IPs and external devices
- Building a modular test suite

Implementation

- MIDI decoder to convert MIDI input to a set of events describing the button presses
- Audio processing layer to convert decoded MIDI input into a digital audio stream
- Input decoder to process information from knobs and rotary encoders
- Video module to display configuration settings on an OLED display

Block Diagram



Testing & Verification

- Software simulators & unit tests for each module in design
- Software simulator of entire MIDI layer, audio layer, and video layer
- Physical verification: verification of latency using high speed audio capture, tuner to measure intonation

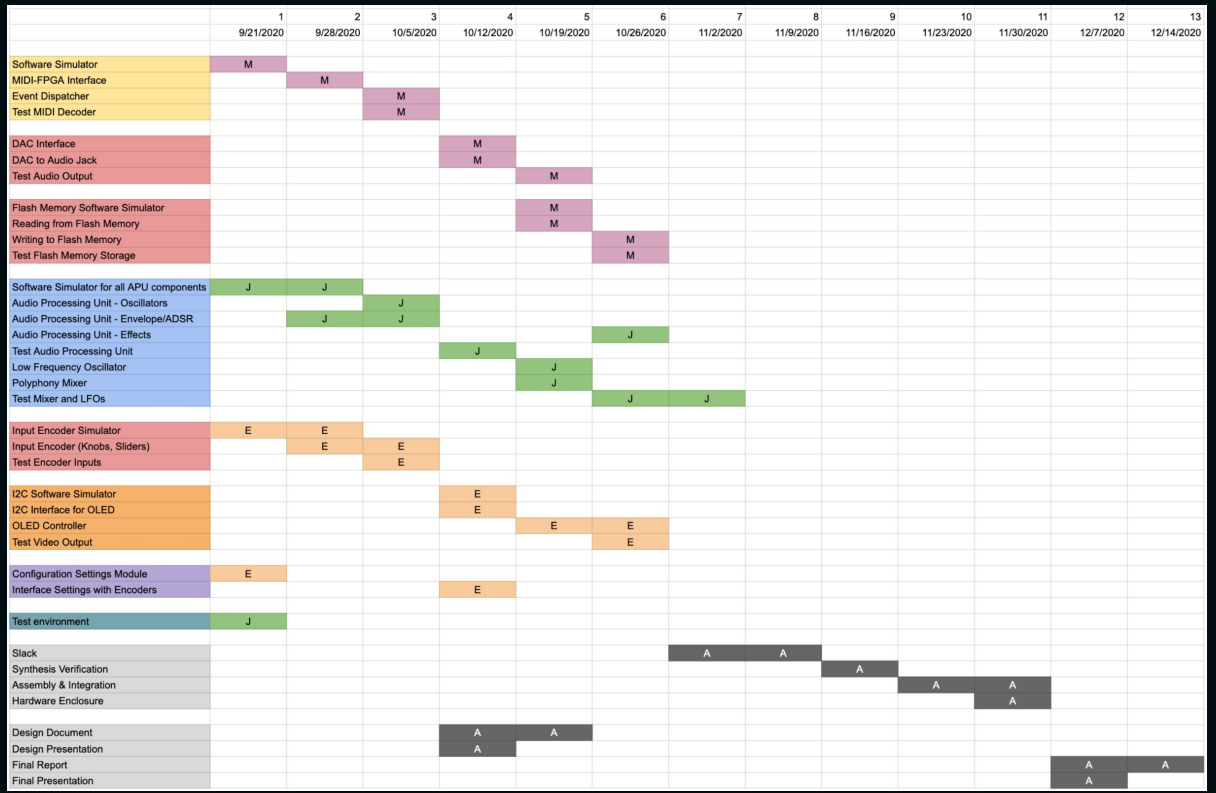
Metrics

- Accuracy of effects and frequencies
  - Can be compared directly to our software models
- Power/area
  - To achieve 4 note polyphony
- Timing
  - To achieve 48KHz 16 bit audio
- Price
  - To make our product significantly cheaper than competitors

Tasks and Division of Labor

Manav	Joe	Eric
• MIDI-FPGA Interface • Event Dispatcher • Reading from Flash Memory • Writing to Flash Memory • DAC Interface • DAC to Audio Jack	• Audio Processing Unit - Oscillators • Audio Processing Unit - Envelope/ADSR • Audio Processing Unit - Effects • Low Frequency Oscillator • Polyphony Mixer	• Input Encoder (Knobs, Sliders) • I2C Interface for OLED • OLED Controller • Configuration Settings Module

Schedule



	1 9/21/2020	2 9/28/2020	3 10/5/2020	4 10/12/2020	5 10/19/2020	6 10/26/2020	7 11/2/2020	8 11/9/2020	9 11/16/2020	10 11/23/2020	11 11/30/2020	12 12/7/2020	13 12/14/2020
Software Simulator	M												
MIDI-FPGA Interface		M											
Event Dispatcher			M										
Test MIDI Decoder			M										
DAC Interface				M									
DAC to Audio Jack				M									
Test Audio Output					M								
Flash Memory Software Simulator					M								
Reading from Flash Memory					M								
Writing to Flash Memory						M							
Test Flash Memory Storage						M							
Software Simulator for all APU components	J	J											
Audio Processing Unit - Oscillators			J										
Audio Processing Unit - Envelope/ADSR		J	J										
Audio Processing Unit - Effects						J							
Test Audio Processing Unit				J									
Low Frequency Oscillator					J								
Polyphony Mixer					J								
Test Mixer and LFOs						J	J						
Input Encoder Simulator	E	E											
Input Encoder (Knobs, Sliders)		E	E										
Test Encoder Inputs			E										
I2C Software Simulator				E									
I2C Interface for OLED				E									
OLED Controller					E	E							
Test Video Output						E							
Configuration Settings Module	E												
Interface Settings with Encoders				E									
Test environment	J												
Slack							A	A					
Synthesis Verification									A				
Assembly & Integration										A	A		
Hardware Enclosure											A		
Design Document				A	A								
Design Presentation				A									
Final Report												A	A
Final Presentation												A	

Conclusion

We will design a system that allows real time frequency modulation and effects to be applied to a standalone MIDI keyboard.

Thanks for your attention!