PROJECT AND REQUIREMENTS

- **FPGSZR is a digital hardware synthesizer with an FPGA as the center of its computation, being able to output a variety of waveforms coupled with effects and filters commonly present in low to mid-range synthesizers.**

Use-Case Requirements	Design Requirements
Lightweight interface for easy transport: < 5 lbs	• Laser cut enclosure with ⅛ inch plywood material • Enclosure dimensions do not exceed 24 x 12 x 6 inches
Short setup time: < 5 minutes	• Plug-and-play MIDI connection (no driver installation required) • FPGA boot time of < 10 seconds

PROJECT AND REQUIREMENTS (CONT.)

Use-Case Requirements	Design Requirements
System I/O Speed: End-to-end Latency of < 5 ms	• MIDI End-to-End Latency of < 2 ms • 24-bit Fixed-Point Signal Representation
Audio Quality: SQNR > 40dB	• Phase accumulator resolution of >= 24 bits • 24-bit Audio CODEC Stereo via I2S • THD < -50dB and SFDR > 50 for sine waves
Audio Accuracy: Pitch accuracy within 10 cents	• Phase accumulator resolution of >= 24 bits
Polyphony: Four voice polyphony	• 4-voice polyphony does not exceed 10ms of total latency

INITIAL SOLUTION APPROACH

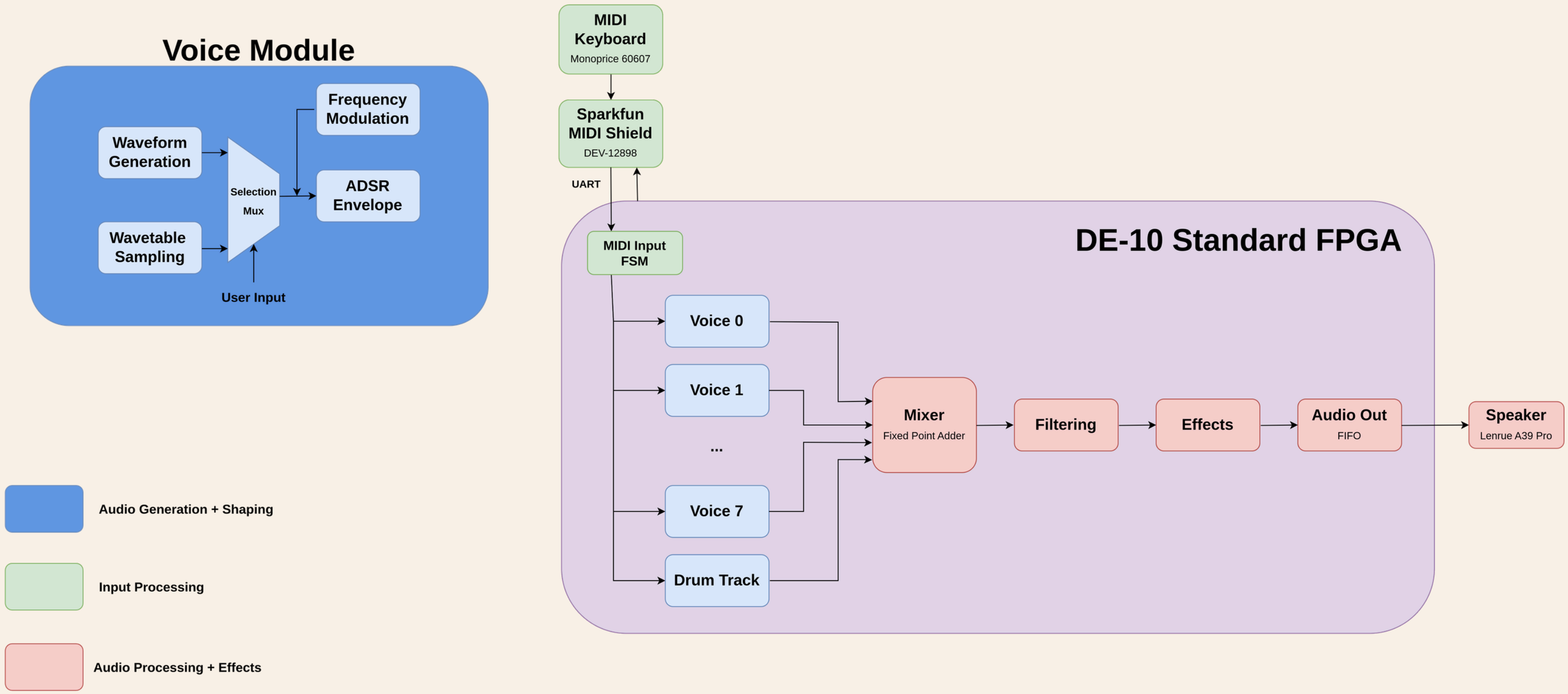

FIGURE 1: ORIGINAL SYSTEM LAYOUT

COMPLETE SOLUTION

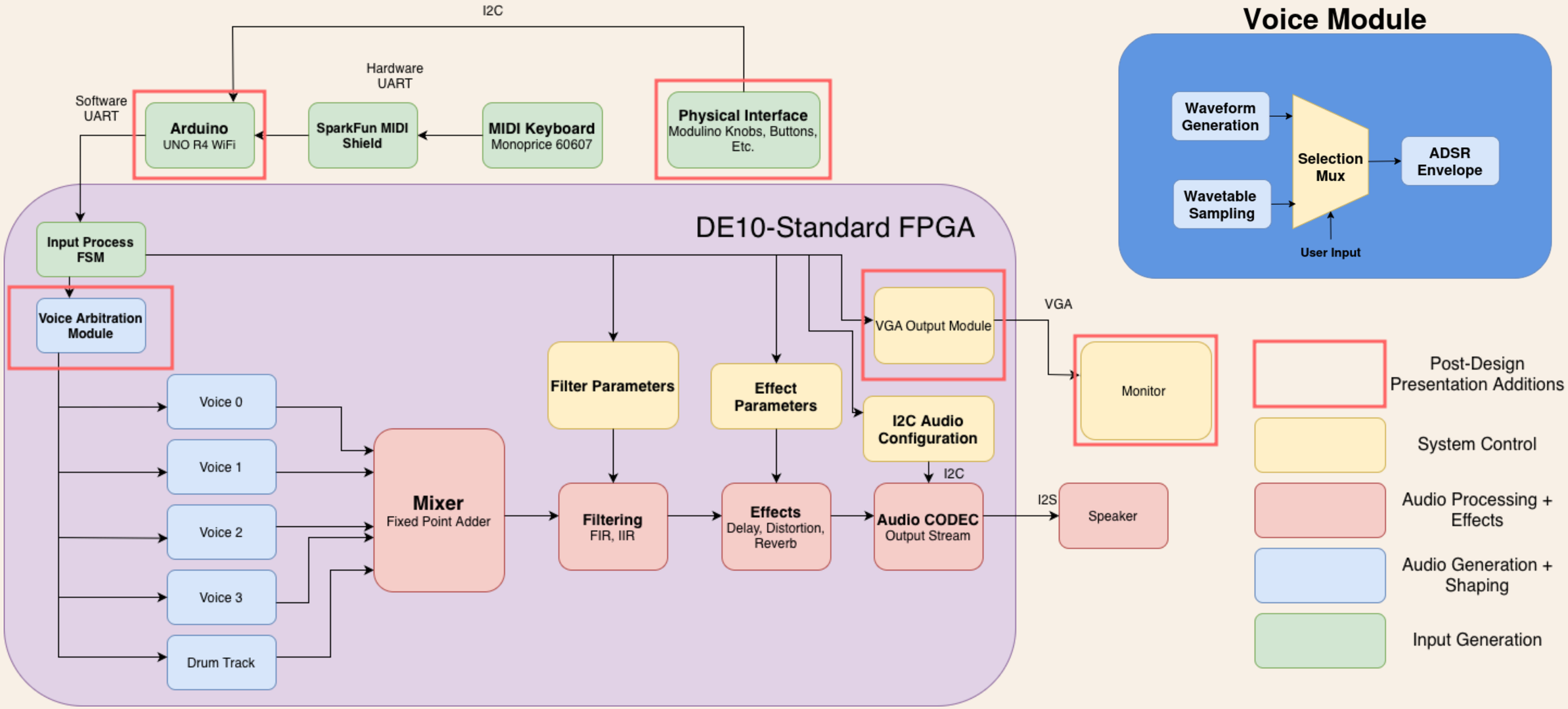

FIGURE 4: COMPLETE SYSTEM LAYOUT

VERIFICATION AND VALIDATION – EXPLICIT TESTS

Metric	Test Method	Succesful Output
System I/O Speed	• Use oscilloscope to capture time difference between initial MIDI bit sent and final bit captured by FPGA.	• < 2 ms for MIDI keyboard input • < 5 ms for overall end-to-end input
Audio Quality	• Capture wave output in .txt file and extract metrics through built-in Matlab functions.	• > +40 dB SQNR for sine waves • < -50 dB THD for sine waves • > +50 dB SFDR for sine waves
Audio Accuracy	• Capture audio output through dedicated cable and input it into a spectrum analyzer to get frequency value.	• < 5 cent difference average for supported notes

VERIFICATION AND VALIDATION – EXPLICIT TESTS

Metric	Test Method	Succesful Output
Interface Weight	• Measure chassis weight with all components inside it using a scale	• < 5 lbs for entire system
Setup Time	• Test 20 users by timing their setup of the system using a stopwatch	• < 5 minutes for setup
Chassis Dimensions	• Measure dimensions of chassis and its material utilizing a tape measurer at TechSpark	• Enclosure dimensions do not exceed 24 x 12 x 6 inches • Plywood thickness does not exceed ⅛ of an inch

VERIFICATION AND VALIDATION – CORRECTNESS TESTS

Metric	Test Method	Succesful Output
Filter and Effect Correctness	• SystemVerilog test bench for subportion of system • Modeling of fixed point arithmetic using python script	• Model output sufficiently close to floating point, and matches SV TB
Input Correctness	• Utilize hex displays on FPGA to output whichever note or knob is being used	• Key or knob value matches expected value from keyboard
Signal Correctness	• Capture output of signal from phase accumulator in a SystemVerilog testbench on a .txt file • Use a ideal signal on Matlab to compare with outputted signal • Compare signal audio outputs to expected outputs	• Testbench output matches Matlab model and audio output matches expected output

LATENCY

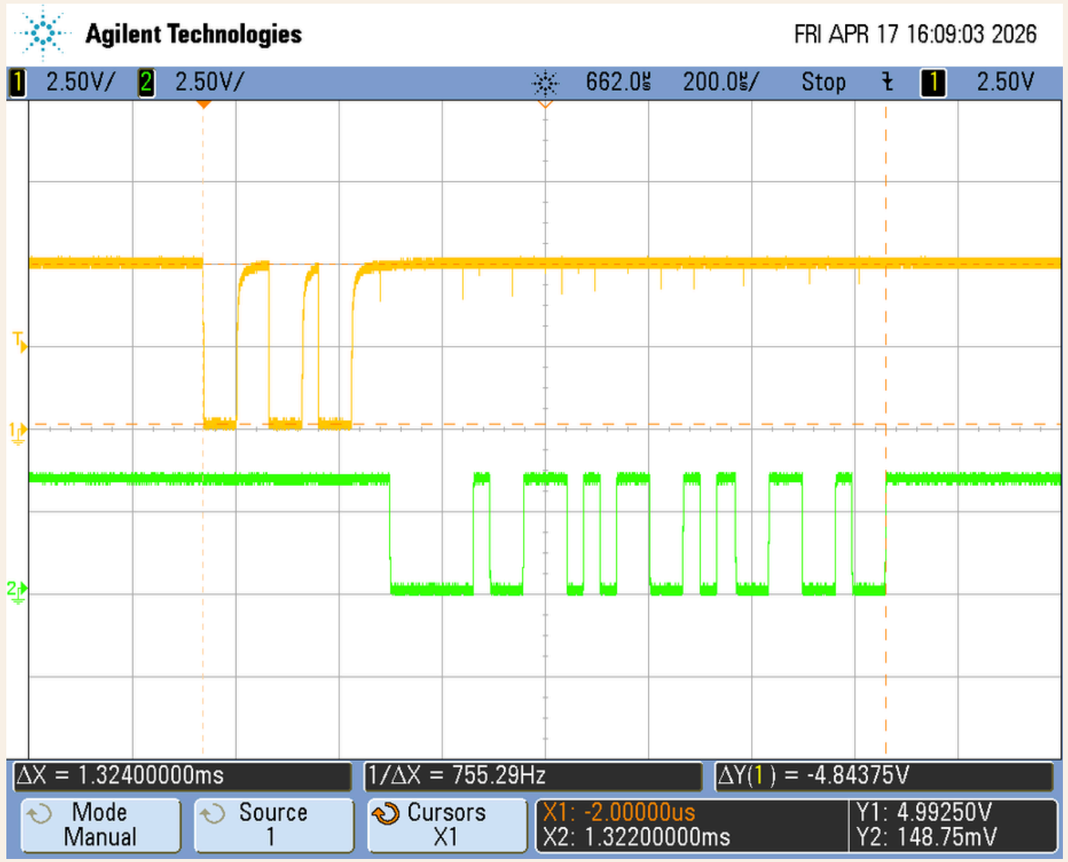

FIGURE 2: OSCILLOSCOPE OUTPUT

Metric	Expected Measurement	Extracted Measurement
MIDI End-to-end Latency	< 2 ms	1.324 ms
Overall System Latency (Singular Note)	< 5 ms	2.426 ms
Overall System Latency (Four Simultaneous Notes)	< 5 ms	2.741 ms
FPGA Boot Time	< 10 s	4.39 s

Meets system spec Does not meet system spec

NOTE: SMALLER LATENCY = BETTER

AUDIO QUALITY

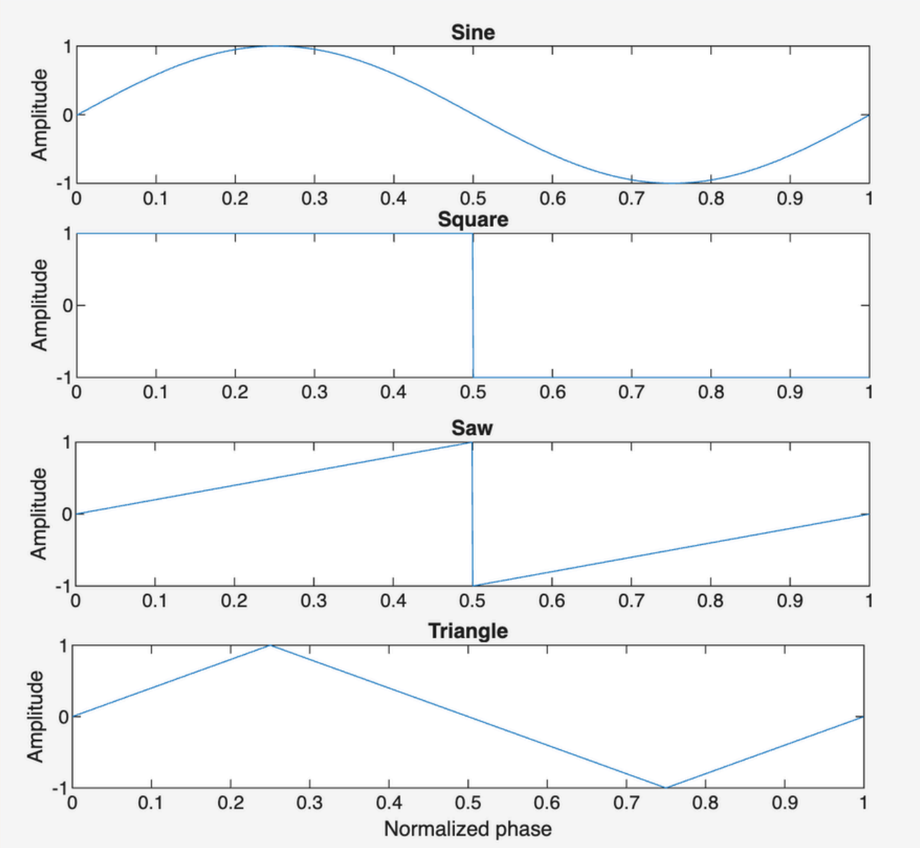

FIGURE 3: WAVETABLE OUTPUT IN MATLAB

Wave Type	THD (dB)	SQNR (dB)	SFDR (dB)
Sine Wave	-55.29	+44.99	+59.32
Square Wave	-6.31	+30.10	+9.54
Saw Wave	-1.90	+25.33	+6.02
Triangle Wave	-18.33	+108.37	+19.08

Meets system spec

Does not meet system spec

SQNR/SFDR: HIGHER = BETTER

THD: LOWER = BETTER

AUDIO ACCURACY

Note	Measure	Ideal (Hz)	Diff
C3	130.8	130.81	-0.13
C#3	138.6	138.59	0.12
D3	146.9	146.83	0.83
D#3	155.5	155.56	-0.67
E3	164.9	164.81	0.95
F3	174.6	174.61	-0.1
F#3	185	185	0
G3	196	196	0
G#3	207.6	207.65	-0.42
A3	220	220	0
A#3	233.1	233.08	0.15
B3	246.9	246.94	-0.28
C4	261.8	261.63	1.12
C#4	277.2	277.18	0.12
D4	293.6	293.66	-0.35
D#4	311.1	311.13	-0.17
E4	329.9	329.63	1.42
F4	349.2	349.23	-0.15
F#4	370	369.99	0.05
G4	392	392	0
G#4	415.2	415.3	-0.42

Frequency Difference: $1200 \times \log_2(F_{measured} / F_{ideal})$

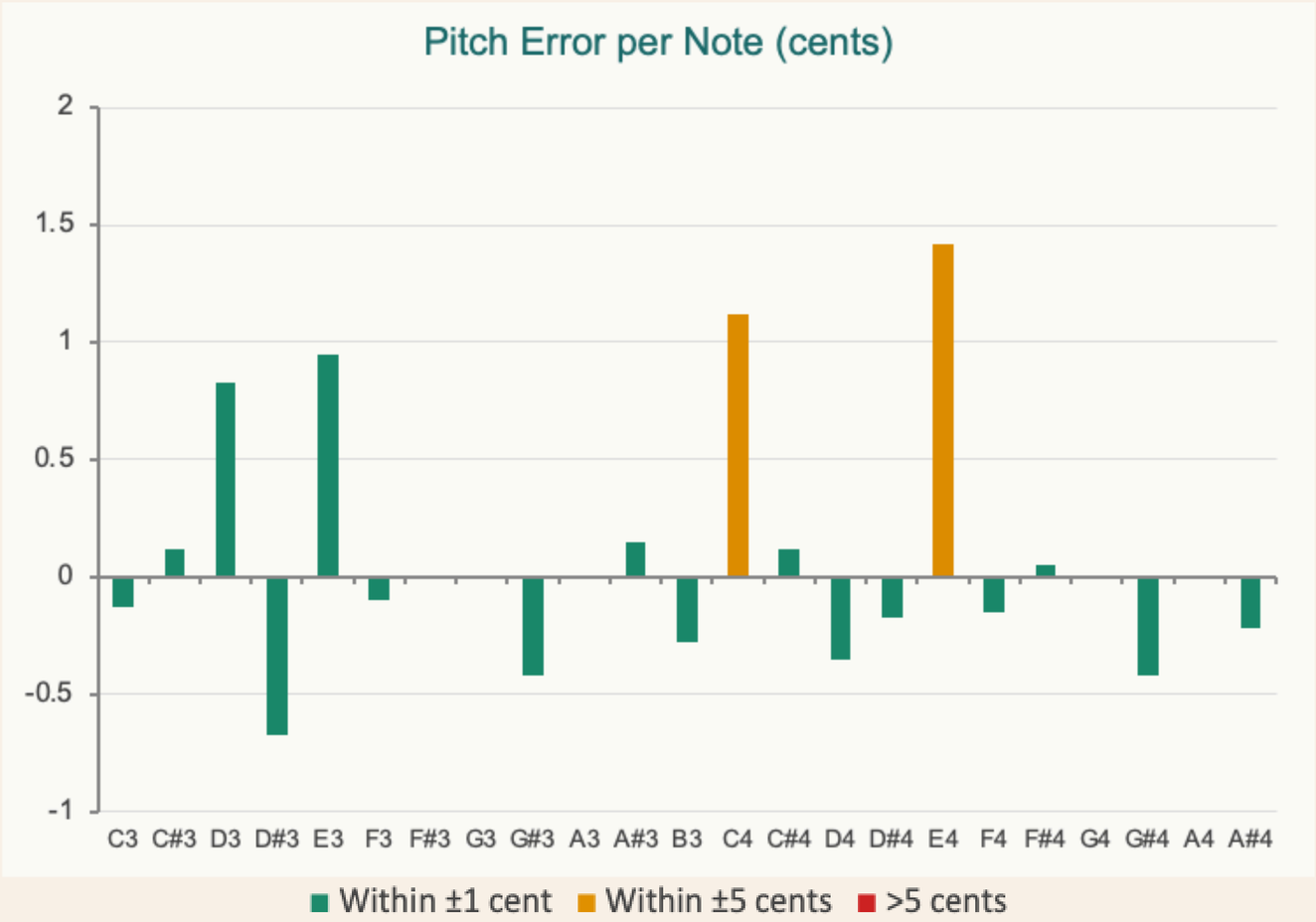

FIGURE 5: PITCH ERROR PER NOTE METRICS

AVG ERROR: 0.33 CENTS ✓

SYSTEM TRADEOFFS

USER INTERFACE

- Decided to perform all interfacing through Arduino than on FPGA
- Using Arduino caused higher latency, but was much easier to program and upscale

AUDIO PROCESSING

- Eliminated Frequency Modulation and Multi-Band Compression
- Reduced functionality in exchange for higher quality in audio output

PROJECT DESIGN

- Decided to laser cut box rather than 3d print
- Increased onboarding time but much cheaper, saving budget

				January	February				March				April			
WBS Number	Task Title	Task Owner	Pct of Completion	1/26-2/01	2/02-2/08	2/09-2/15	2/16-2/22	2/23-3/01	3/02-3/08	3/09-3/15	3/16-3/22	3/23-3/29	3/30-4/5	4/6-4/12	4/13-4/19	4/20-4/26
1	Logistics and Setup															
	Website Setup	Everyone	100%													
	Slack	Everyone	100%													
	Proposal Presentation	Everyone	100%													
	Design Presentation	Everyone	100%													
	Design Report	Everyone	100%													
	Interim Demo	Everyone	100%													
	Final Presentation	Everyone	100%													
	Poster PDF	Everyone	0%													
	Final Video	Everyone	0%													
2	Project Planning															
	Plan Out Design/Block Diagrams	Everyone	100%													
	Order Materials	Everyone	100%													
	Finalize Overall Design	Everyone	100%													
3	Project Definition															
	DAC I2C Audio Setup Integration	Jose	100%													
	DAC I2S Audio Setup Integration	Jose	100%													
	DAC Integration with Waveforms/Testing	Jose, Kaloyan	100%													
	MIDI Shield Setup/UART Design	Jose	100%													
	MIDI Simulation Debugging	Jose	100%													
	MIDI Physical Debugging	Jose	100%													
	Knob and Button Interface Design	Jose	100%													
	Interface Testing/User Box Design	Jose	100%													
	Audio and Interfacing User Testing	Jose	0%													
	Waveform Generation Design	Kaloyan	100%													
	Waveform Generation Integration	Kaloyan	100%													
	ADSR Filter Design	Kaloyan	100%													
	ADSR Filter Implementation/Test	Kaloyan	100%													
	Drum Playback Integration	Kaloyan	100%													
	Distortion Effect Design/Integration	Kaloyan	100%													
	Reverb Effect Design/Integration	Kaloyan	0%													
	IIR Filter Design	Mihail	100%													
	FIR Filter Design	Mihail	100%													
	Delay Effect Design	Mihail	100%													
	Freq Modulation Design	Mihail	100%													
	Delay Effect Testing	Mihail	100%													
	IIR/FIR Filter Integration/Testing	Mihail	100%													
	Overall Filter Testing/Integration	Mihail	100%													
	Polyphony Mixing Filter Design	Mihail	0%													
	Overall System Integration	Everyone	40%													