

Perfect Pitch Music Maker

6.111 Final Project Checklist

Adam McCaughan and John O'Brien

Whole System

- Functionality: Take audio input of monophonic music from a microphone and user inputs of time signature and clef. Convert the audio information to sheet music format shown on the display.
- Demonstration: Play a recording with a known sheet music representation into the microphone, compare the resulting display to the sheet music.

Digital Fourier Transform

- Functionality: Converts 21ms of audio data into the frequency domain every millisecond
- Demonstration: VGA output of frequency/magnitude plot for an internally generated pure tone

Tone Converter

- Functionality: Intelligently converts information from the frequency domain into a single characteristic frequency
- Demonstration: Input of a variety of artificially generated frequency maps, confirmation of output as correct characteristic frequency. Output is to the hex display.

Tone LUT

- Functionality: Takes a characteristic frequency and outputs corresponding pitch and octave.
- Demonstration: Input a range of frequencies using the hex display and the labkit buttons. Output the pitch and octave to the hex display. Check that the frequency is mapped to the correct pitch and octave.

Score Converter

- Functionality: Checks current pitch and octave every $1/32$ of a beat and if necessary reports the previous values of pitch and octave to graphics control along with a duration value.
- Demonstration: Artificially input pitches and octaves, check to make sure the module reports proper fields for score element (including duration, pitch, octave and start time)

Graphics Control

- Functionality: Takes pitch, octave, duration, start time and a ready signal and produces context dependent instructions on how to draw three

32x140 pixel slices. Once the recording operation is finished allows the user to flick through the pages of recorded score.

Demonstration: If the artist module and frame buffer are both operational, this module can be demonstrated by inputting a series of score elements from a ROM. If these modules are not functional then n0, n1, n2 and graphics_ready can be displayed on the hex display.

Artist Module

Functionality: Takes specific instructions on how to draw three 32x140 pixel slices and outputs a stream of the required pixels and coordinates to the frame buffer.

Demonstration: Faked values of n0, n1, n2 and graphics ready can be inputted from a ROM. If the frame buffer is functional then the correct elements can be observed on the screen. If the frame buffer is not functional one slice at a time can be demonstrated using the hex display.

Frame Buffer

Functionality: Takes pixel values and associated coordinates and stores them to a buffer. At the same time it outputs a stream of pixel values to the xvga module. In this way it allows the display to be incrementally updated.

Demonstration: Fake a flashing block input to the module that can be moved with the up, down, left, right buttons on the labkit. Demonstrate that the buffer 'remembers' the value of the flashing block.

If time permits...

Add additional methods of tone detection to improve accuracy in determining the and octave.

Add a playback module that will synthesize the recorded piece. This would be demonstrated by recording a piece then playing back a synthesized version