CSEE 4840 - Embedded Systems
Project Proposal: Piano Heroes

Anita Bui-Martinez (adb2221), Michael John Flynn (mf3657),
Zakiy Tywon Manigo (ztm2106), Robel Wondwossen (rw3043)

Spring 2025

1 Overview

Our team intends to create a video game inspired by *Piano Tiles - Don't Tap the White Tile* created by Hu Wen Zeng in 2014. The game will have a series of black and white "piano" tiles scrolling down from the top of the screen. Users will use an attached piano keyboard to play the corresponding keys in the correct order before they scroll off the screen. When each key is pressed, a piano note will be played.

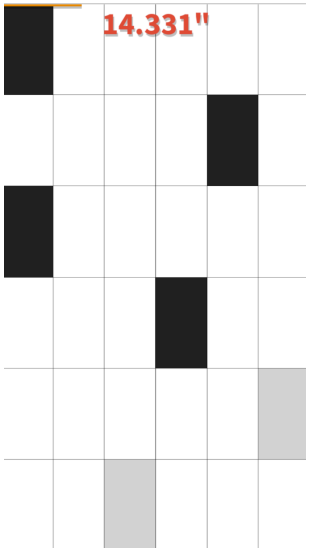

Figure 1: Piano Tiles Display

2 Design

The game will display tiles scrolling down the screen while taking input from the USB keyboard to control how the user is responding. Based on the user input, the game logic will decide whether or not to continue the game or to end it if the user has made an error.

The design will be broken into the following components:

- **Display Module** - The VGA display will render the scrolling tiles in real time. They will fall at a constant rate with the potential to have their speed increased for more difficult levels or game modes. The display will also update based on user input, such as grayed-out tiles when correctly pressed, the score or time updating, and a game over icon when the user taps the wrong key or misses a tile.
- **Input Module** - The USB keyboard will detect input strokes and send information to the software to ensure that the right keys are being pressed at the correct times.
- **Audio Module** - Pressing a key triggers a sound to play, which would be the corresponding piano note for that song.
- **Game Logic Module** - This will handle the progression of the game, score increasing or decreasing, time advancement, and checking that the user is playing the correct input. It will also ensure that the tiles are scrolling correctly and not accounted for once pressed or off screen. The logic will also potentially account for different game modes or corresponding songs.

If time permits, we hope to implement different game modes, such as zen mode, arcade mode, or choosing specific songs from a catalog.

3 Hardware-Software Components

3.1 Software

Our software will consist of:

- **Game logic** - takes care of input validation, scoring, game progression
- **Input handling** - takes input from the keyboard and translates them into actions in the game
- **VGA Driver** - renders the graphics and buffer
- **Audio Driver** - responsible for playing the corresponding piano key sounds

3.2 Hardware

Our hardware will consist of:

- **FPGA Board** - takes care of processing, rendering graphics, and handling input
- **USB Keyboard** - captures input from the user and sends it to the software to interpret them. If time permits and/or we have issues with a prebuilt keyboard, we may try to make it ourselves.
- **VGA Display** - Outputs the scrolling tiles for the game.
- **Audio Output** - Generates the game's sounds

4 Tasks and Challenges

1. Interface Components - make sure all components, such as the keyboard, FPGA, display, and audio are all communicating and working together as intended
2. Render Graphics - ensure a smooth scrolling effect
3. Audio Implementation - Ensure that the correct sound is played with minimal delay
4. Game logic - make sure the game is responsive and working smoothly
5. If time permits, add additional features, such as different modes, expanded song selection, additional graphics.