Rhythm Master

Group member: Yangyang Zhang (yz4843), Junfeng Zou (jz3850), Hongrui Huang (hh3084)

1. Project Overview

Target Platform:

DE1-SoC FPGA, VGA Display, USB Keyboard, Speaker

Project Outline:

Develop a rhythm-based music game where players press keys in synchronous with falling notes. Core features include game menu, music play, dynamic note scrolling, hit feedback, and scoring system. Extended features add complexity and customization.

2. System Architecture

FPGA (SystemVerilog):

- Implement core game logic, including note timing, scoring, and hit detection.
- Process and store note timing data for real-time judgment.
- Manage synchronization between gameplay events and audio play.
- Handle music play.

HPS (C):

- Handle global game settings and controls.
- VGA rendering via frame buffer updates.
- USB keyboard input processing.
- Communicate with FPGA via the Avalon-MM bridge for note status and score updates.

3. Basic Functionality Implementation

3.1 Game Menu System

Main Menu:

- Options for "Start Game", "Tutorial", "Settings", "Exit", "Difficulty/Song Selection" (optional)

Tutorial Screen:

- Visual guide to gameplay mechanics.

Settings:

- Adjust volume, screen calibration, etc.
- Manage game-wide settings such as difficulty and control preferences.

In-Game Pause Menu:

- Pause the game via a key press, showing options: Continue, Restart, and Return to Main Menu.

3.2 Gameplay Mechanics

Music Play:

- Drive an external speaker by onboard audio codec chip (WM8731) on FPGA

Note Rendering (VGA):
- Draw 4 or more tilted "lanes" (tracks) on the screen
- Generate falling notes from the top, synced to the music's beat.

Hit Feedback:
- Define a "judgment zone" at the bottom of each lane.
- Detect key presses (via USB keyboard) and compare timing with note positions.
- Provide visual feedback based on timing accuracy: "Perfect", "Good", or "Miss".

Scoring System:
- Single note hit score is based on timing accuracy (10/5/0 points).
- Combo hits earn extra points, and display the combo number and current score on the screen.
- After a piece is played, give the player an aggregate score and the level classification (S, A, B and C), trying to give the players a goal to achieve to make them more engaged.

4. Advanced Features

4.1 Long Notes (Hold Notes)
- Render elongated notes requiring players to hold keys.
- Track key press duration and deduct points for late presses or early releases.

4.2 Difficulty Adjustment
- Adjust the difficulty of the same song by adding more notes or speeding up the notes scrolling.

4.3 Song Selection
- Store multiple tracks on the SD card, modify the menu for selecting songs with previews.

4.4 Audio Hit Feedback
- Try to implement playing song accompaniment and playing the note when the player presses down the key, which involves FPGA DSP and audio overlay.

5. Milestones

5.1 Music Play and VGA Rendering
5.2 Game Menu System
5.3 Hit Feedback and the Scoring System
5.4 Advanced Features Implementation (Achieve as much as possible)