ECE 3375B
Microcontroller Design Interim Report

Andrew Randell 250911270
Robert Potra 250914807
Gautam Gupta 250897104
Yunyi Zhang 250908830

March 18, 2019

Contents

1 Scope/Introduction

This project will implement a bespoke guitar hero style game on the six 7-segment displays with the DE10-standard board. This project will demonstrate a complete understanding of low-level hardware interfacing in C with ARM processors.

1.1 Problem Definition

This project is classified as a game. The objective of a game is to entertain people. Thus, there is no large environmental or political problem solved with this project.

2 Functional Description

2.1 System Purpose

The system will re-create the popular arcade game - guitar hero on the DE10-standard board. The game will be played on 3 different layered tabs on segments A, G, and D, of the 6 displays, see Figure 2. The built in LCD display on the DE10-standard will be used for real-time score keeping.

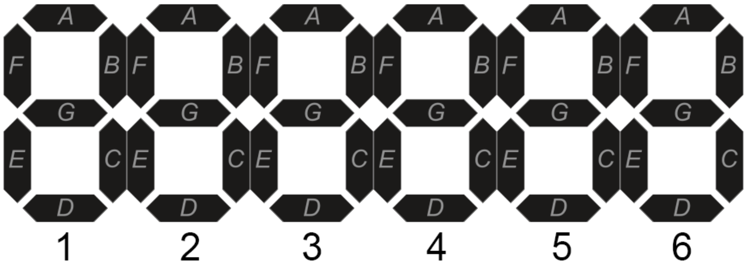

Figure 1: 7-segment Display Array

2.2 User Interaction

The user interface consists of an LCD display, all 7-segment displays, 3 push buttons, and two switches. The user primarily interacts with the system using the two switches to start/pause the game and to toggle between feedback and current score. To play the game, the user presses buttons used to represent the 3 horizontal segments A, G, and D for displays 1-4. The object of the game is for the user to press the correct button when the horizontal segment is transitioning from display 4 to 5. If the user does not press the correct button at the correct time, the horizontal segment continues to displays 5 and 6.

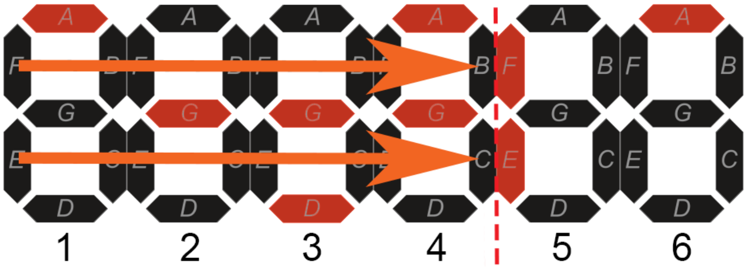

Figure 2: 7-segment Display User Interface

2.3 Devices Used

In the prototyping phase of this project, no external devices or systems will be used. The DE-10 components are sufficient to construct an effective prototype. The following DE-10 components will be used:

- LCD display
- 7-Segment display
- 3 Push buttons
- 2 Switches
- HPS Timer Module

2.4 Quantities Monitored and Stimuli

The number of tabs hit on time, early, or late are tracked to calculate the score. The number of tabs hit in a row will also be tracked to count for the longest tabs hit in succession for a combo. The visual stimuli that affects the user input will come from the 7-segment display as that is where the game play will occur and the LCD score display. Section 3.2 goes into detail about the button layout in accordance to the display stimuli.

3 I/O Requirements

The system will utilize the buttons as binary inputs from the user. The user will determine when to press these buttons based on the tab outputs from the 7-segment displays. The system will also implement switches with one of the switches being used to start the game and the other switch being implemented to toggle the LCD display to either show the score of the current session or the feedback of how accurate the users button input was. A final switch will be used to show the longest combo in the session.

3.1 Seven-segment Displays

The game will be played on the six 7-segment displays. The third display will be the player's fingers; the vertical segments F and E on display 5 (see Figure 2) will always be illuminated to indicate where notes must be hit. Notes will be displayed coming from the leff on horizontal segments A, G and D on displays 1 thru 4; any missed notes will proceed past the player's fingers and onto displays 5 and 6 indicating a missed note.

3.2 Button Inputs

The following push buttons in Figure 3 map to the segments in Figure 2 according to the following structure:

$$W \to 5A$$

$$X \to 5G$$

$$Y \to 5D$$

$$Z \to Idle$$

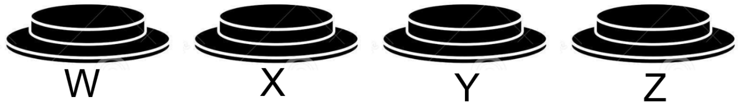

Figure 3: Push Buttons

3.3 LCD Display

The LCD will display the number of hits and misses in the current level and can be toggled to show the current score. The score data will include the number of notes hit, the number of missed notes, and the elapsed time since the last missed note.

3.4 Switch Inputs

Switches will be used to control the information displayed on the LCD display and the start of the game. When the first switch is in one position, the display will show the player's score. When the first switch is in the other position the display will show the player if the last button pressed was too early, at the correct time, or late. The second switch will be used to start or pause the game.

4 Initial Software Design

4.1 High-level Software Control

4.1.1 Note Generation

For the first iteration, notes will be generated at random and spawned on display 1. A random number generator will be used to determine which line to place the notes on. A second iteration of this game would intake an MP3 file, decompose the frequencies into three categories, and spawn notes according to the current frequency which would be emitted from a speaker. This step would more-closely resemble playing a guitar.

4.1.2 Timing

The HPS timer will be used to time the game. The timing of the game will start off slow and increase as the game progresses to increase difficulty.

To determine whether the user has hit or missed a tab, each tab will have a thresh hold on how much time the tab will accept to be considered a hit, early hit, or late hit. If the user inputs outside of the thresh hold, the game registers it as a miss for the closest note according to time.

4.1.3 Score and Combo Tracking

At every successful note hit, a global counter (score) is incremented and displayed on the LCD display. If multiple notes are successfully hit in succession, then a second global variable (combo) is incremented. The combo variable is reset to 0 after the user fails to hit a note.

4.2 Low-level Hardware Interfacing

Low-level C functions will be written to interface with all hardware-level components on the board. The main hardware interactions consist of the following:

4.2.1 Timing Mechanism

The HPS timer will be used to determine hits, early hits, late hits, and misses.

4.2.2 Game Display

The 7-segment display will be programmed to take in Hex values

4.2.3 Score Display

The LCD display will be programmed to show either the current score, current combo, or the accuracy (hit, early, late, miss)

4.3 High-Level Local Testing

The components stated in subsection 4.1 will be tested prior to being tested with the hardware in the labs. From previous DE-10 assignments, the process of displaying numbers on the 7-segment display have been reliable. The array that holds the hexadecimal to 7-segment display will be changed to accommodate for the new patterns that have to be changed, such as the notes, and the feedback in response to user inputs. The timing from the HPS and its according accuracy can be displayed in the computer terminal as simulation for when it will be displayed on the LCD display. The score and combo tracking can be also be tested on the computer terminal.

5 Prototyping Plan

This project is entirely contained on the DE10-standard board. Thus, there is no hardware design required. The software will be written in C and Github will be used for version control. A project timeline is available in Figure 4. Testing will be completed during lab periods.

5.1 Timeline

Figure 4: Timeline

Week of:	
Mar. 11	**Problem Definition and I/O Requirement** • Brainstorm Ideas • Complete progress report
Mar. 18	**High-level Software Design** • Refine details included in progress report • Simulate the algorithms on local computers
Mar. 25	**Low-level Firmware Design** • Apply the algorithm and logic of the game onto the board's display • Implement the algorithms variables as inputs on the board (switches and buttons)
Apr. 01	**Implementation and Testing on DE10-Standard** • Implement and debug the code onto the DE-10 board in the lab times • Explore possible additional features on the board