Guitar Hero: Fast Fourier Edition

A 6.111 Final Project by Mitchell Gu and Ryan Berg
The Guitar Hero Sensation

- Harmonix founded in 1995 at the MIT Media Lab and eventually developed the first Guitar Hero game
- The genre became wildly popular, resulting in a wide array of games.
Limitations of the old Guitar Hero model

- Lacking in realism
- Lacking in song flexibility
- Limited education value
Why our idea solves these problems

- Uses an actual guitar, rather than a guitar-shaped controller
- Actual guitar tablature (a common form of writing music for guitar) is played, instead of an arbitrary 5-note model of music.
- The only difference between our game and actually playing a guitar is the addition of a graphic interface and scoring.

The interface for rhythm-based guitar games

An example of actual guitar tablature
Motivation

The two of us both love playing guitar and want to build on the entertainment value of current guitar-based rhythm games while also making guitar accessible to more people.

This project is also a great opportunity for us to learn about working in the frequency domain, developing concurrently, and integrating many systems together.
High-level Project Design
Low-level Implementation
Audio Processing + Note Recognition on the labkit

Target range: E2 to E5
(3 octaves, 12 half-step each for 37 total notes)
Corresponding frequencies: 87 Hz to 659 Hz
Serialization and Deserialization: Labkit to Nexys4

Diagram:
- **LABKIT**
  - Serializer
  - NSYNC
  - NDATA (Serial)
- **LABKIT**
  - Deserializer
  - NDATA
  - 36
  - 36
Note matching on the Nexys 4
Graphics modules on the Nexys 4

The diagram shows the AV generation system with various modules including:

- Score Graphics
- Audio Out Handler
- Game/Background Graphics
- Graphics Integrator
- String Graphics x6
- Pause Menu Graphics

Connections include score, fret, time, and pause inputs, with outputs for PWM Audio Out, VGA Out, and the score display.
Game Control Logic on the Nexys 4
# Timeline

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Scoring Module</td>
<td>M</td>
<td>M</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2) FFT/Note Recognition</td>
<td>M</td>
<td>M</td>
<td>M</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3) Functional AV</td>
<td></td>
<td>R</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4) Serialization/deserialization</td>
<td></td>
<td></td>
<td>M</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5) Game Control Logic</td>
<td></td>
<td>R</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6) Integration, testing</td>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>7) Completed SD Card Song/Metadata</td>
<td></td>
<td></td>
<td></td>
<td>R</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8) Buffer Time/ Stretch Goals</td>
<td></td>
<td></td>
<td></td>
<td>A</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9) Project demo, final checkoff</td>
<td></td>
<td></td>
<td></td>
<td>AAAHHH</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Testing

Since we are using both a labkit and a Nexys 4, we can concurrently develop modules and test each of them to be more time-efficient.
Resources

We need two 3.5mm male -> 1/4in female audio connectors (~$3 each on Newegg). We both already have guitars, amps, cables, etc.

We need the prebuilt Verilog for a Labkit FFT, for reading/writing to an SD card with the Nexys4, and for generating PWM audio output with the Nexys4.