1.0 Motivation

Final check-off consolidates your entire project into a complete digital storage oscilloscope. From the Video Encoder, SDRAM, SDRAM Arbiter, Waveform Generator, and AC97 Audio Controller, this “checkpoint” is where the rubber hits the road and your complete project comes to life.

From a digital design perspective, this checkpoint’s primary goal is to prove the value of interface driven design to you. Throughout the project, you have been augmenting your modules with the Ready/Valid interface. By this time, a bird’s eye view of your project is a bunch of different pieces, along with an SDRAM Arbiter, that feature this interface. Typically in CS150, the processing of integrating the final project into a complete design, from its different components, is highly non-trivial. This semester, the work you put into maintaining the Ready/Valid interface will ensure that connecting the pieces is just that: connecting the pieces.

Functionality-wise, you will be required to demonstrate the following for final check-off:

1. Record audio data from:
   a. A waveform generator (WaveformSource.v from checkpoint 4).
   b. A microphone (through the audio codec).
2. Play that data back out to both:
   a. The TV as a waveform.
   b. The TV, as an actual sound, through the speakers (through the audio codec).

This list of requirements can be reworded into the following requirements:

1. Waveform Generator (checkpoint 4) \(\rightarrow\) TV as a waveform (checkpoint 4)
2. Waveform Generator (checkpoint 4) \(\rightarrow\) Audio through speakers (checkpoint 5)
3. Microphone (checkpoint 5) \(\rightarrow\) TV as a waveform (checkpoint 4)
4. Microphone (checkpoint 5) \(\rightarrow\) Audio through speakers (checkpoint 5)

The rest of this document will discuss these requirements in more detail.
3.0 Prelab

1. Finish checkpoints 1-5 of your project.

2.0 Datapath / Requirement

Recall from the Project Specification the below block diagram.

![Block Diagram]

Figure 1 Top-level Project Block Diagram

Note that the Waveform Generator has been added to this diagram. Also note that there is an “Ethernet Audio” block that makes up a third possible input into the Arbiter. This requirement is now extra credit (you will NOT have to make it work for final check-off). We will keep it in the diagrams for when we discuss it as an extra credit option.

2.1 Record Audio from a Variety of Sources

Final check-off requires that you are able to record audio from a waveform generator (WaveformSource.v from checkpoint 4) and from the audio codec (checkpoint 5). To master this requirement, the following pieces (highlighted in red) must be functioning correctly.
1. The audio controller (you **checkpoint 5** solution in the Serial → Parallel direction).
2. The SDRAM Arbiter (**checkpoint 3**).
3. The SDRAM Controller (**checkpoint 2**).
4. An address counting and muxing scheme capable of dealing with the 2 clients (waveform generator, audio codec).

![Diagram of Audio Recording Requirements](image)

**Figure 2 Audio Recording Requirements**

The only requirement for final check-off is that recording between multiple clients “works.” The actual implementation decisions are up to you. For example, you might decide to actually mux multiple clients through a single Write FIFO into your Arbiter. Alternatively, you might decide to extend your Arbiter to support more Write FIFOs and disregard the mux entirely. The choice is up to you. **The requirement only stipulates that you are able to record audio from various sources.**

**Note that the TAs will not check-off this section of the project stand-alone.**

We cannot verify that you are recording data correctly without seeing it displayed or hearing it played somehow. As such, this requirement exists only so that you know what sources you will be required to support, and that you actually have to record audio and not just pass it through. Credit for satisfying the above requirements comes in the next section, where you will actually display (through sound or image) what you have recorded.
2.2 Display / Play Audio

With audio from some source written to SDRAM, your check-off becomes massaging that audio into different forms so that it can be seen and/or heard. **For each source, you will be required to display recorded audio on the TV as a waveform and played through the TV’s speakers as pure sound.** See the check-off sheet (on the back of this specification) for details. Note that this requirement includes playing WaveformSource.v as a sound. WaveformSource.v will be augmented with a pure tone sound of an audible frequency so that it doesn’t sound like garbage. Assuming your means of recording data is in place, the blocks involved with display are shown in the diagram below.

![Diagram of Audio Display Requirements](image)

**Figure 3 Audio Display Requirements**

The TA who checks you off will ask you to do the following:

1. Record sound from the microphone on command.
2. Play that sound back through the speakers on command.

Note that your solution should be able to perform both of these actions “on command” or at the click of a button. **Your solution can’t merely play back sound at some predetermined time after it is “recorded.”**

After your TA has verified that your recorded audio can be played back through the speakers, you must demonstrate that your solution can…
3. Display the sound that was played out through the speakers as a waveform on the TV screen.
   a. Scroll through the sound (fast-forward and rewind from checkpoint 4).
   b. Display the sound playing in real-time (real-time mode from checkpoint 4).

Again, this should be “on command” or configurable through buttons, dipswitches and the like (the decision is up to you – you must just be able to make the transitions). Note that your solution should be able to switch back and forth between playing sound and displaying waveforms. Also note that playing sound through the speakers and displaying it on the screen as a waveform can be decoupled for final check-off. You will only be required to show that one (not both) works at any given instant in time. Playing back audio and displaying it as a waveform at the same time will be added as extra credit (see the Extra Credit section of this specification).

Once your TA verifies the above for recorded audio through a microphone, you will be required to demonstrate the exact same functionality for “audio” recorded from WaveformSource.v.

### 3.0 Extra Credit

As you know, extra credit is due along with your final project as a part of final check-off. The judging for the project competition will not take place during final check-off. Final check-off will be used by the staff to take note of what extra credit you have finished so that we can assign credit (that contributes to your course grade). Note the following:

1. You can still get extra credit if you do not manage to complete all of the requirements of final check-off. Do not put off final check-off to complete extra credit, however. Final check-off is worth 20% of your project grade.
2. Your extra credit must be integrated into the same .bit file as the rest of your final check-off submission. It is implied, therefore, that your extra credit cannot break the project’s base offering.

Below are two additional extra credit options that will be added to the Extra Credit specification.

#### 3.1 Lab6 Audio Playback (1.5%)

- **Requirements**
  - Add the ability to record audio from your lab 6 solution.
  - Your implementation must be able to perform all of checks listed above (in verifying the display from other sources).

- **Note on Streaming Audio**
  - The TAs will try as best as possible to keep the audio stream live throughout the next week.
  - As with lab 6, audio will go down occasionally when no TAs are around to restore it.
If you plan on implementing this option, have something else to work on when the audio stream is down.

3.2 Couple Waveform-Playback Functionality (1.5%)

- Requirements
  - Change playback so that in addition to playing back sound or displaying a waveform at different times, you can also playback-display at the same time.
  - The waveform that shows on the TV should resemble the sound that you hear from the speakers (and the two should not fall out of sync as time progresses).
4.0 FINAL CHECK-OFF

**Assigned:** Monday, November 17th

**Due:** Wednesday, November 26th, 11:59pm sharp

<table>
<thead>
<tr>
<th>Man Hours Spent</th>
<th>Total Points</th>
<th>TA Initial</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>/ 100</td>
<td>/ / 08</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Final check-off is worth more than any other checkpoint. Additionally, there is a special check-off procedure in place. Please read the Project Specification for more details.

I. Waveform Generator: Displays correctly on TV
   
II. Waveform Generator: Played back over the Speakers
   
III. Checkpoint 5 Audio: Shown as a Waveform on the TV
   
IV. Checkpoint 5 Audio: Played back over the Speakers

---

RevA – 11/14/08       Chris Fletcher       Wrote new document.