ECE 350: Digital Systems Final Project Proposal

Dylan Peters, Justin Havas, Noah Pritt

November 3, 2018

Duke Community Standard

By submitting this LaTeX document, I affirm that

1. I understand that each `git` commit I create in this repository is a submission
2. I affirm that each submission complies with the Duke Community Standard and the guidelines set forth for this assignment
3. I further acknowledge that any content not included in this commit under the version control system cannot be considered as a part of my submission.
4. Finally, I understand that a submission is considered submitted when it has been pushed to the server.

1 Overview

Our idea is to create a messenger application that allows people to chat from one FPGA to another, connected via ethernet. This project will start out as a simple chat application, and we will add extensions to increase the types of communication that are possible. For example, from the basic text chat application, we could expand to allow sending of photos, voice calling, and even video chat.

2 Inputs and Outputs

The FPGA should be able to read inputs from a keyboard using the PS2 protocol we learned about in lab. The FPGA should be able to process the keyboard inputs and format them as data packets to be outputted via the ethernet cable. The FPGA should then be able to read the input from the ethernet cable (sent from the other FGPA), and process the letters that were sent. Finally, the

FPGA will need to output visuals onto a screen via VGA in order to display the message that was received.

If we are able to add further extensions, we would like to add voice and video calls. In order to implement these, the FPGA will need to be able to take sound input via the "Audio In" port and output sound via the 35 mm headphone jack. Finally, the FPGA will need to take input from the "Video In" port, process that data, send it across the ethernet, and the FPGA on the other side will need to receive and display that data.

3 Tasks

1. Read input from a keyboard via PS2 protocol
   (a) This is necessary in order for the user to be able to type messages.
   (b) This will be of mild difficulty, as we have not yet done PS2 protocols.
   (c) We will use the processor to decide what to do with the letter that was typed.
   (d) Input: Signal from PS2 keyboard
   (e) Output: Letter that was typed
   (f) Processor Use: Moderate
   (g) Points: 10
2. Display text on screen via VGA
   (a) This will allow the user to see what is being typed and received.
   (b) The difficulty of this is based on the fact that we will have to define how the letters should look and create images for each letter, which should be displayed next to each other neatly.
   (c) Input: Letter that was typed on keyboard or received via ethernet
   (d) Output: Text on screen via VGA
   (e) Processor Use: Significant
   (f) Points: 20
3. Write data to ethernet port
   (a) This is necessary in order to communicate with the other FPGA.
   (b) This will be very difficult because we need to ensure that the FPGAs are not both writing at the same time.
   (c) Input: Letter to be written to ethernet
   (d) Output: Signal on ethernet cord
   (e) Processor Use: Moderate
   (f) Points: 30

4. Read data from ethernet port
    (a) This is necessary in order to communicate with the other FPGA.
    (b) This will be very difficult because we need to ensure that the are synchronized so they can read the data from the other FPGA, which may have a clock cycle slightly offset from its own.
    (c) Input: Signal on ethernet
    (d) Output: Letter that was received
    (e) Processor Use: Significant
    (f) Points: 30
5. Read input from a microphone and encode it to be sent to other FPGA
    (a) While we have already read microphone data, this task involves manipulating the data in order to send it to the other FPGA. This will likely require the processor in order to format the data to be sent.
    (b) Input: Microphone data
    (c) Output: Signal on ethernet cord
    (d) Processor Use: Moderate
    (e) Points: 10
6. Read microphone data from ethernet and play it
    (a) By this point we will have learned how to read ethernet data, so the challenge will be to decode the microphone data from the text message data. This will require a good data encoding scheme that allows us to determine what each packet is for.
    (b) Input: Signal on ethernet cord
    (c) Output: Speaker signal
    (d) Processor Use: Significant
    (e) Points: 10
7. Read input from a video camera and encode it into ethernet cord
    (a) This will require us to learn how videos are encoded, decode the data from the camera, and re-encode it onto the ethernet channel.
    (b) This will be extremely difficult, as we have not discussed video in this class, and the camera data will be much denser than the microphone or keyboard data.
    (c) Input: Video camera data
    (d) Output: Signal on ethernet cord
    (e) Processor Use: High
    (f) Points: 30

8. Read input from ethernet cord and display video on screen
    (a) This will require us to decode the data sent over the ethernet cord and display it in real-time, in order to allow two people to chat. This will be an extremely difficult task because of the amount of data being processed and received over the ethernet port.
    (b) Input: Signal on ethernet port
    (c) Output: Signal on VGA port to be displayed
    (d) Processor Use: High
    (e) Points: 30
9. Apply dimming effect via FPGA switches
    (a) We want to allow the user to control the brightness of the on-screen text produced by the FPGA. To do this, we will use the switches on the FPGA as dimmers. When all the switches are at 1, then the image will be bright. When they are all 0, it will be dark, and when some of them are 1 and some are 0, it will be somewhat dim and somewhat bright.
    (b) Input: Switch positions
    (c) Output: VGA signal
    (d) Processor Use: low
    (e) Points: 10
10. Use FPGA buttons to produce animations.
    (a) We want to allow the user to create on-screen animations using the buttons on the FPGA. For example, the user could press one of the buttons, and the screen would display confetti superimposed on top of the text messages.
    (b) This will require color calculations when the animation is superimposed above the text.
    (c) Input: Button presses
    (d) Output: VGA signal with animations superimposed over background
    (e) Processor Use: low
    (f) Points: 10
11. Save chat to SD card to persist between power cycles
    (a) We want to allow the user to continue chatting with someone after a reboot or loss of power to the FPGA. We can achieve this by saving the chat log to an SD card.
    (b) This will require a lot of processor commands to calculate the proper addresses to save to and load from.

(c) This will also be a very difficult task because nobody has ever been able to save to SD cards before, so we will need to determine how to read to and write from it.

(d) Input: Chat log (stored in memory)

(e) Output: Chat log saved to SD card

(f) Processor Use: Very High

(g) Points: 30

4 Timeline

PC 6: November 8	Implement tasks 1 and 2. Type on a keyboard and see the text display on screen via VGA.
PC 7: November 15	Implement tasks 3 and 4. Read and write data between two FPGA boards via ethernet.
Between PC 7 and PC 8: November 22	Implement tasks 5 and 6. Record audio via microphone and send over ethernet. Read data from second FPGA board and play audio.
PC 8: November 27	Implement tasks 7 and 8. Use a video camera to record video and send over ethernet. Read data from second FPGA board and play video.
Final submission: December 6	Implement tasks 9, 10, 11. Apply dimming effect via FPGA switches, use FPGA buttons to produce animations, and save chat to SD card to persist between power cycles.