ECE 385

Final Project Proposal

Adam Urish and Jason Zou

(adamwu2, jasonz3)

4/9/21

Proposal Overview

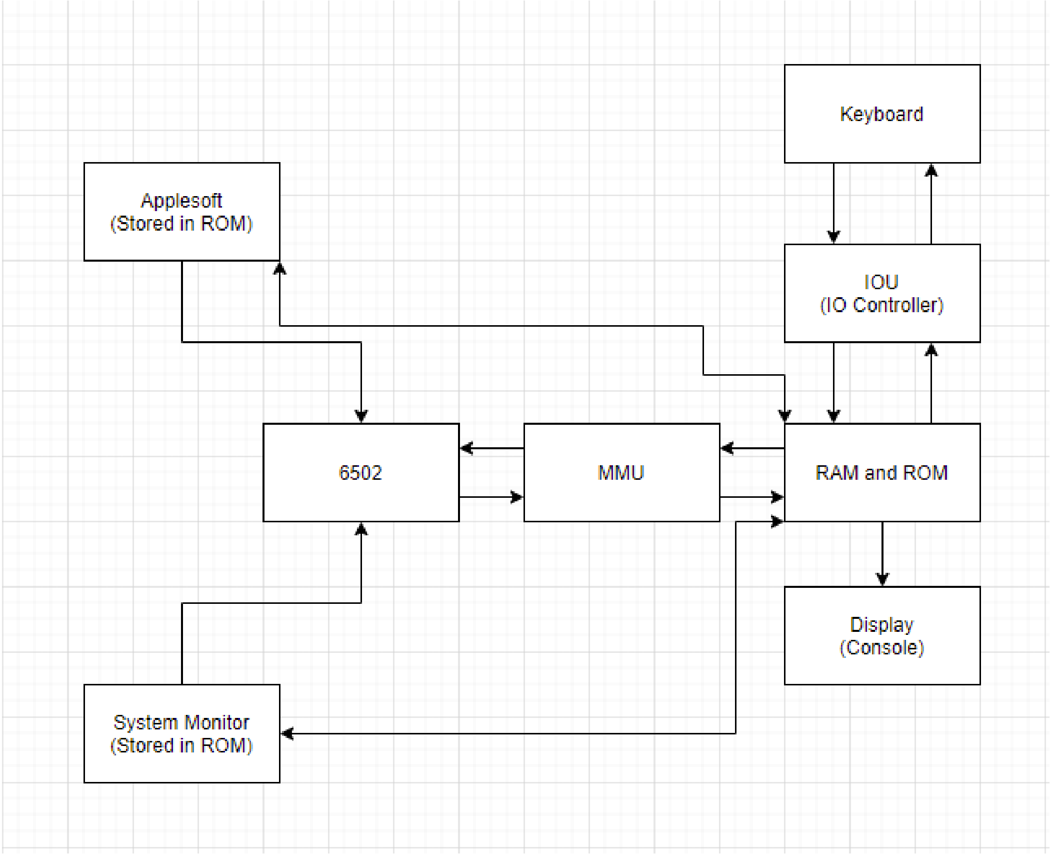
Our proposal entails implementing an Apple IIe with the DE-10 lite FPGA board. We will be using code for 6502 CPU from an existing source [LINK]. For the other components of the system, they will be implemented in SystemVerilog. The drivers required for the FPGA shield to work with a keyboard and an SD card will be written in C. The ROM for the Apple IIe will be aquired externally [LINK].

The expected difficulty for this project will be very challenging. This is due to the requirement of implementing software and ensuring it functions properly on hardware it was never designed for.

For our timeline, we plan to have the 6502, memory bank, and text output working by the midpoint check in. This will allow us to show some basic functionality of the Apple IIe, perhaps by displaying data from a set region in memory. From this base, we will work on the keyboard input next, then the ROM programs, System Monitor and Applesoft. At this point we will have a relatively functional Apple IIe, and we can begin on the advanced features, like graphical output, getting a game running, getting the SD card to act as a floppy disk, and running DOS.

Block Diagram

The reference manual includes several block diagrams, but they are very verbose and show every connection. We will certainly use them during our project, but for this proposal I have created a very simplified diagram.



Feature List

Baseline

- Functional 6502 CPU in SystemVerilog
- Keyboard input
    - Will be using a NIOS-II instance to map USB keyboard input to Apple keyboard input
    - Keyboard plugged in through the FPGA Shield
- Text Output
    - In 40 and 80 column text output mode, the display data is stored as ASCII bytes in memory, so we can pull this and display it directly in a console.
- Functional Memory Bank
    - This will involve creating SV files for the MMU, as well as the memory itself.
    - Three main sections
        - Main RAM
        - Memory Mapped I/O
        - Bank Switching memory
            - Can switch between ROM and extra RAM
- System Monitor
    - From the manual: “The monitor is a computer program that is used to operate the computer at the machine level”
    - Built in firmware subroutines for accessing memory, I/O, etc.
    - Stored in ROM
- “Applesoft”
    - Apple IIe built in BASIC interpreter
    - Stored in ROM
    - A BASIC interpreter can be used to demo some text based games if we are unable to get graphical output working

Advanced

- Some form of a game
    - Oregon Trail, Donkey Kong, Pacman, etc.
    - Will be loaded off the SD card using the NIOS-II instance to convert them to floppy disk type format
- Apple DOS
    - Apple “Disk Operating System”
    - Also loaded off of a floppy disk
- Graphical output
    - Apple IIe supports graphical output at a resolution of up to 280x192
    - Done thru memory mapping, would have to translate to vga signal.
- **(VERY ADVANCED)** We have a physical 6502 chip from a previous class, and it would be cool to see if we could somehow hook it to the DE-10 and use it for the Apple IIe. We will only attempt this if we finish everything else, as it will likely be quite difficult and might require parts we do not have.