EE/CprE/SE 491 WEEKLY REPORT 7

Digital ASIC Fabrication

3/27/2023 – 4/2/2023

Group number: sddec23-06

Client & Advisor: Dr. Henry Duwe

Team Members/Role

- Jake Hafele
- Gregory Ling
- Will Galles
- Cade Breeding

Past Week Assignments

ALL

- Keep working on developing rtl models for each of the internal modules.

Will

- Work on developing a detailed block design for the DSP module.

Jake

- Create a detailed block design for the SPI module including shift registers and data buffers for clock synchronization
- Discuss how to handle clock synchronization between external master SPI and slave SPI module in the user area

Meeting Notes from Previous Week with Dr. Duwe

3/27/23

Updates from us:

- Created powerpoint of top level module specifying I/O of each submodule
  - Easiest thing to do is associate wires with names

        - Show figures with arrows going from one module to another
        - Show power and/or clock signals colored separately
        - questions to think about:
            - Whats the flow of data between the modules?
            - Whats in the user area, whats not?
            - Whats in the wishbone bus?
        - Show something to a reader not familiar with the project
        - We will want something more similar to the design specification document we outlined last week
    - Will met with Isaac Rex to go over the DSP Road Noise algorithm
        - Updated submodule in design specification
        - Drew out diagram relating user area to microcontroller
        - Could end up being very large
        - Could implement all layers on a piece of hardware
        - Use memory to cache in between
            - Do we need SRAM?
        - How much data do we need to hold onto?
            - Could be massive
        - This will be the largest resource user
        - Isaac also implemented a Weiner filter
            - Tuner to voice/audio that can be inserted directly instead of the neural network
            - Lighter weight if neural network too large
        - Clean data will only be from training
            - Stored as distribution of frequencies
            - LSTM can be type of deep Neural network
        - Performs Fourier transform to get magnitude of different frequency range, adjusted with filter, and rebuilt magnitudes back into audio signal
            - Sequence would be helpful to know
            - Should be able to use LSTM with fixed weights
                - Think Isaac pretrained and copied weights into Neural Network
                - Access to code, all done in MATLAB
                - Would have to retrain network since the weights were deleted
        - Any relevant papers from CPRE 487?
            - Not really, not big focus of the class for LSTM's
        - LSTM's beneficial for audio application because time is involved
        - Goal of performance?
            - Reach real time by sampling 8 kHz
            - 125 uS per inference
        - For sake of design document, calculate for real time, calculate target
            - May have to back off on accuracy if we are close
            - Width and granularity of FFT. What is the bin size?
            - Interpolate samples?
        - No recurrence used?
            - Isaac stated he wanted to explore recurrence in future
            - Isaac advised to move from non-recurrent network to recurrent network for better results
            - Should we instead focus on the one he did do?

        - Or alternatively, focus on the Weiner filter instead
            - Gives us opportunity to look at LSTM in the future, working from a place with all auxillary circuits designed
            - Both will use atleast the FFT modules, and could be reused
    - Clock Speed?
        - Not defined yet
        - What COULD our clock speed be?
        - 2-8 MHz at the low end
        - Configurable clock speed
            - May not be able to run at top speed
            - Wishbone bus clocked at management SOC
            - What is Management SOC clock speed?
    - VISIO schematic
        - Submodules are all slaves to wishbone bus
        - Any interrupts going back into management SOC
            - No, maybe good to do?
            - How else to determine if done? Polling?
            - DSP algorithm? Plan was to insert filtered data and pull from wishbone bus for input every cycle
                - May be easier to raise interrupt flag if data is ready
                - Only problem is feeding it data
                - Two interrupt flags, one data ready, one input empty

TODO:

- Update top level schematic
    - Show connected with arrows
    - Differentiate power and clock signals (different colors)
- DSP Road Noise
    - Look into training
    - How many weights? How large?
    - How much data do we need to store?
    - How many computations to do?

Weekly Progress

- Started to compare the different ways of implementing the Weiner Filter. I believe that it is going to be better to do one massive convolution in the time domain continuously than to perform the FFT on the incoming data. I believe that we will need less memory and space by going the route of direct convolution. - Will
- Researched SPI master/slave clock synchronization. We decided to use multiple DFF's so that the ready indicator after receiving data from the Master SPI module would wait two clock cycles from the user area of our chip, to confirm valid data. -Jake

- Created a block diagram for our backdoor SPI module to determine Verilog implementation. It uses two shift in registers for the data address/read indicator and 32 bit data in from the SPI master. We also included a shift out register to send data from the slave to the master SPI module, and finally some DFF's for a buffer with clock synchronization. -Jake

Pending Issues

- Get an idea about the size of a multiplier to figure out how big the convolution can be or how many stages we would have to break the system down into.
- Determine how cells will be compiled from Verilog. Do we need to structurally instantiate D Flip Flops or will Openlane automatically detect that in our Verilog code?

Individual Contributions

- All contributions were for the group this week.

Team Member	Contribution	Weekly Hours	Total Hours
Jake Hafele	Research SPI clock synchronization, Draw out SPI block design in detail, begin Verilog modules	6	34
Gregory Ling	Project Plan and helping Jake with the clock domain crossing	2	30
Will Galles	Compare different ways of implementing Weiner Filter	5	42
Cade Breeding	Project Plan and starting with implementation of the standard cells	4	30

Plans for Upcoming Week

- Get a (12x12?) multiplier implemented and hardened to get a identify the size needed for the larger DSP project.

- Test using Magic to create a custom logic cell – Gregory
- Create verilog modules for shift in register, shift out register, and backdoor SPI module. Investigate how this will get compiled with Openlane and Skywater standard cells. -Jake

Appendix