FPGA implementation of Cooley-Tukey

Garrett Massman and Cory Walker

September 13, 2014

Overview

In this project we will implement the Cooley-Tukey FFT algorithm on a Xilinx FPGA using VHDL.

Theory

Describe the definitions of the DFT and the Cooley-Tukey FFT.

In Mathematica

Describe Mathematica code, how precomputed tables can help us.

FPGA considerations

Describe how we plan to utilize SPI for communication, what type of logical blocks we will use and how they will work together.

Block diagram

Over the past few weeks we have taken what we know an outlined a basic block diagram of our device:

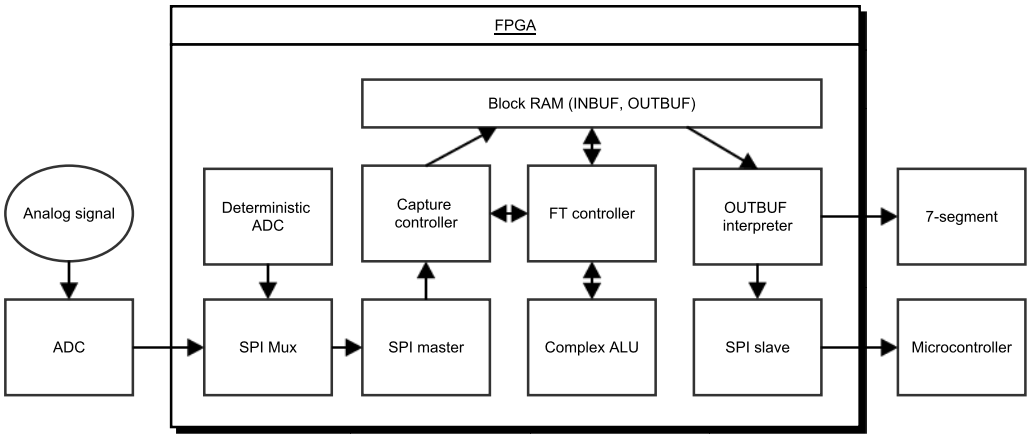

Figure 1: Block diagram of the FFT chip

Now that we have divided the device into several logical blocks, we can now split up the work accordingly.

Roadmap

Describe the steps, in order, that need to be done.

Resources

List of resources that we plan to use.

Possible extensions

Describe the extra things we might add if we have time.

Conclusion

References

[1] Doin, Jonny. *SPI Master/Slave Interface.* OpenCores, 16 May 2011. Web. 13 Sept. 2014. <`http://opencores.org/project,spi_master_slave`>.

[2] Reynwar, Ben. *FFT on an FPGA.* FFT on an FPGA. N.p., n.d. Web. 13 Sept. 2014. <`http://www.reynwar.net/ben/docs/fft_dit/index.html`>.

[3] Satoh, Keiichi, Jubee Tada, Kenta Yamaguchi, and Yasutaka Tamura. *Complex Multiplier Suited for FPGA Structure.* Computers and Communications (2008): 341-44. Web. 13 Sept. 2014.

[4] Wikipedia contributors. *Cooley–Tukey FFT algorithm.* Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 27 Jun. 2014. Web. 13 Sep. 2014.

[5] Wikipedia contributors. *Discrete Fourier transform.* Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 2 Sep. 2014. Web. 13 Sep. 2014.