Wideband Full-Stokes parameterized spectrometer for Xilinx RFSoC Platform

DESIGN DOCUMENT

Team 41
Professor Phillip Jones
Advisers
Team Members/Roles
Brian Bradford
Vishal Joel
Jared Danner
Louis Hamilton
Nick Knuth
Team Email
sdmay19-41@iastate.edu
Team Website
sdmay19-41.sd.ece.iastate.edu

Revised: 10/8/2018/#1.0

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

NOTE: This template is a work in progress. When in doubt, please consult the project plan assignment document and associated grading rubric.

CASPER - Collaboration for Astronomy Signal Processing and Electronics Research

MiGen - Fragmented Hardware Description Language based in python to automate the VLSI design process.

Xilinx - FPGA manufacturer. *The FPGA used in this project is a Xilinx product.

DSP - Digital Signal Processing.

SDR - Software Defined Radio.

FPGA - Field Programmable Gate Array.

SoC - System on a Chip

1 Introduction

1.1 Acknowledgement

CASPER [1], the "Collaboration for Astronomy Signal Processing and Electronic Research" have addressed this problem by developing platform independent hardware and open source software to take advantage of developments in Field Programmable Gate Array (FPGA) and Analog to Digital Converter (ADC) technology and "quickly" target new platforms such as the Xilinx RFSoC evaluation board (ZCU111).

Xilinx University Program - Donation of the Xilinx RFSoC Eval Board.

Alan Langman (Engineer @ Vermeer Corporation) - Technical guidance for Migen, CASPER, and spectrometer related material.

Dan Werthimer (Chief Scientist: SETI@Home) - Project idea, CASPER, and digital signal processing support.

1.2 Problem and Project Statement

The problem our team will be challenged with is overhauling the CASPER Toolflow in order to allow for developers to easily become proficient in using and expanding on the existing tool set. The current toolflow prohibits new developers from making quick progress due to lack of documentation, the need for prior knowledge on MATLAB / Simulink, and an absence of support for other FPGA platforms beyond Xilinx, such as Altera or Lattice. In doing so, our team will also develop the board support package for the Xilinx RFSoC Eval Board.

This project will work toward porting of the CASPER spectrometer instrument and its dependent libraries to the Xilinx RFSoC evaluation platform. The team will also look at improving the board support package by migrating the current tools to use Migen, an open source python library for generating and building gateware projects [4]. The team will also work on extending the toolchain to support the Xilinx Pynq infrastructure [5], which will be interfaced to the existing CASPER Python libraries.

1.3 Operational Environment

The end product will be operating within already operational server racks that are climate regulated to match manufacturer specification of optimal operating conditions including: temperature, humidity, and dust ingression. Therefore, the end product does not need to

be built to work in non-common conditions due to its surrounding climate being externally controlled.

1.4 Intended Users and Uses

CASPER [1] is a community of hundreds of scientists and engineers around the world, who collaborate on the development of radio astronomy instrumentation. The CASPER community has reached out to developers to integrate Xilinx's new RFSoC platform into their current tools. Once supported, anyone making use of the open-source tools will be able to quickly develop digital instrumentation for their specific applications.

Should time allow, we will begin the next iteration of the UC Berkeley SETI Research Center's Search for Extraterrestrial Radio Emissions from Nearby Developed Intelligent Populations (SERENDIP) program [3]. This installment would be the next generation instrument system for the Search for Extraterrestrial Intelligence (SETI) coined SERENDIP VII. The system will be an open-source, ultra-high resolution, wide-bandwidth dual-pol spectrometer to be used on the world's largest radio telescopes.

1.5 Assumptions and Limitations

At this point in time, our team is assuming that basic device drivers have been developed and tested. That the CASPER spectrometer instrument will be nearly 100% compatible and operational on the Xilinx RFSoC Platform.

1.6 Expected End Product and Deliverables

Our expected deliverable is a Board Support Package for a Xilinx Zynq Ultrascale+ RFSoC evaluation board. The board support package includes porting for the CASPER spectrometer and its dependent libraries for the Xilinx board. The CASPER tool flow is to be migrated to support Migen, an open source python library for generating and building gateware projects . It should support Xilinx Python drivers to interface with existing CASPER libraries. We must include full documentation of the implementation of the board support package. This end product is to be delivered by the end of Senior Design in May 2019.

2. Specifications and Analysis

2.1 Proposed Design

Include any/all possible methods of approach to solving the problem:

– Discuss what you have done so far – what have you tried/implemented/tested, etc?

– We want to know what you have done

• Approach methods should be inclusive of **functional and non-functional requirements** of the project, which can be repeated or just referred to in this section

If your project is relevant to any **standards** (e.g. IEEE standards, NIST standards) discuss the applicability of those standards here

Successful creation of a Board Support Package file. This includes proper integration of the existing toolflow and any new tools to create a working Board Support Package for the Xilinx Zynq RFSoC platform. Migen will be used as the generation tool for our project which will include ensuring that it is both integrated in the current tool flow and a new tool flow of our own design.

Proper documentation of process. We will need to document our process so that the open source project work can be continued by others. We also need to document our process to allow for review by the CASPER project and to give a tutorial of how our updated tool flow works.

Proper documentation of tools used. Proper documentation of our work and use of Migen is important for expanding Migen itself. We also need to document how the existing toolflow and its tools integrate with Migen.

Functional requirements:

- CASPER spectrometer support
    - The Xilinx board must be made to support the CASPER Spectrometer and the libraries it is dependent on.
- Migen
    - Migen must be able to send script to CASPER's Vivado to produce projects that the FPGA can use.
- Spectrometer
    - Parameterized inputs for user to quickly build

Non-Functional requirements:

- Backwards compatibility
    - The ability to STILL use Matlab/Simulink and existing CASPER tool flow, instead of Migen to program the board.
- Open source
    - Anyone can use our code in their application.

CASPER operates entirely under the GNU General Public License V2.0. This grants the use of the source code to anyone for commercial or private use. This also means anyone has the right to modify and/or distribute the code, however CASPER accepts no liabilities or warranties on said source code.

2.2 Design Analysis

– Discuss what you did so far

– Did it work? Why or why not?

– What are your observations, thoughts, and ideas to modify or continue?

– If you have key results they may be included here or in the separate "Results" section

-Highlight the **strengths, weakness**, and your observations made on the proposed solution.

CANNOT ELABORATE ON THIS UNTIL LATER.

3 Testing and Implementation

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or a software library

Although the tooling is usually significantly different, the testing process is typically quite similar regardless of CprE, EE, or SE themed project:

1. Define the needed types of tests
2. Define the individual items to be tested
3. Define, design, and develop the actual test cases
4. Determine the anticipated test results for each test case 5. Perform the actual tests
6. Evaluate the actual test results
7. Make the necessary changes to the product being tested 8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you've determined.

3.1 Interface Specifications

– Discuss any hardware/software interfacing that you are working on for testing your project

We will be testing Migen and Pynq python libraries. Migen has built in testbenches and simulation for each module. We are also testing the feasibility of the ZCU**111.**

3.2 Hardware and software

– Indicate any hardware and/or software used in the testing phase

– Provide brief, simple introductions for each to explain the usefulness of each

- Hardware

    - ZCU**111** - Final Development board for the project. Contains each of the necessary components on board for the CASPER project.
    - ZCU**10**6 - Intermediate board to test preliminary Migen implementations and work out possible Migen to VIvado issues.
    - Pynq Board - For testing purposes only, offered by Professor Jones.
- Software
    - Migen - Primary development software. Both frontend and new toolflow backend.
    - Vivado - Xilinx software that will build the output from Migen and allow for tweaks if errors are found.
    - Matlab/Simulink - Old toolflow fortend that we still plan on supporting.

3.3 Functional Testing

We will be unit testing the DSP, ADC, and Network interface components. We will also be system testing the board as a whole. We will have to integrate test each component as it is completed to maintain functionality. Finally, we will have to test the system in a real world spectrometer application.

3.4 Non-Functional Testing

Test the accuracy of the DSP data flow and calibration of the ADC. We will also have to benchmark the performance of the 1 gigabit ethernet core and the potential **10** gigabit. We will be testing the compatibility of the Migen code we develop with the old toolflow. We also need to test the usability of our interface for the new toolflow.

3.5 Process

– Explain how each method indicated in Section 2 was tested

– Flow diagram of the process if applicable (should be for most projects)

CANNOT ELABORATE ON THIS UNTIL LATER.

3.6 Results

– List and explain any and all results obtained so far during the testing phase

- – Include failures and successes
- – Explain what you learned and how you are planning to change it as you progress with your project
- – If you are including figures, please include captions and cite it in the text

  • This part will likely need to be refined in your 492 semester where the majority of the implementation and testing work will take place

-**Modeling and Simulation**: This could be logic analyzation, waveform outputs, block testing. 3D model renders, modeling graphs.

-List the **implementation Issues and Challenges**.

CANNOT ELABORATE ON THIS UNTIL LATER.

4 Closing Material

4.1 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

We haven't reached a point where we are testing or even beginning development on most of the components above. As a consequence we are unable to determine if our path is the best solution or if any alternatives are available.

4.2 References

[1] CASPER - http://casper-dsp.org/

[2] "Spectrometers and Polyphase Filterbanks in Radio Astronomy", Danny C. Price, May 2018, https://arxiv.org/abs/1607.03579

[3] SERENDIP VI - https://ieeexplore.ieee.org/document/7436240/

[4] Migen reference - https://m-labs.hk/migen/index.html/

[5] PYNQ reference - https://readthedocs.org/projects/pynq/

[6] SERENDIP VI - https://ieeexplore.ieee.org/document/7436240/

4.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar that does not directly pertain to the problem but helps support it, include that here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc. PCB testing issues etc. Software bugs etc.