Capstone Project Proposal

Team 9 - Lattice FPGA

2025/1/15

Team: Mohammad Alshaiji <alshaiji@pdx.edu>

Riley Cox <rilco@pdx.edu>

Nathaniel Fraly <nfraly@pdx.edu>

Haoyang Han <haoyang@pdx.edu>

Xiang Li <shane29@pdx.edu>

Version 0.1

Table Of Contents:

1. Executive Summary

The goal of this project is to develop a low-power FPGA-based system that drives an SPI-based LCD display. The system will enable users to adjust brightness and cycle through colors using two buttons, showcasing the flexibility of FPGA solutions. Our team will implement three low-power design techniques and compare the outcomes of using an Open Source FPGA toolchain with the Lattice Radiant toolchain. This project emphasizes FPGA advantages in efficiency and cost-effectiveness over microcontrollers while addressing industry challenges like power consumption.

The deliverables include a fully functional prototype housed in a 3D-printed enclosure, supported by Verilog code, comparative analysis reports, and visualizations of power consumption. This product will serve as a demonstration tool for Lattice Semiconductor, emphasizing the potential of FPGAs in low-power, cost-sensitive applications. Development will occur over three phases, ensuring a robust and scalable outcome.

2. Background and Research

Lattice Semiconductor is an FPGA company Based in Hillsboro, Oregon. They market themselves as the low-power programming leader. According to their SEC 10-Q filing on November 04, 2024, their total assets as of that date totaled $853,661,000. Their total revenues for the quarter ending in September 2024 was $127,091,000. From their investor relations statements, they describe themselves as "[solving] customer problems across the network, from the Edge to the Cloud, in the growing communications, computing, industrial, automotive, and consumer markets."(Lattice)

FPGAs serve a particular niche of the market concerned with parallel processing, high-speed processing and the ability to reconfigure hardware. Such properties are leveraged by communication, automotive, aviation, and artificial intelligence systems among others. Traditionally, FPGAs stand as an alternative to the use of ASICs and microcontrollers. For this project specifically, the sponsor is interested in challenging the contention that FPGAs consume more power, are more expensive, and necessitate more space than microcontrollers by implementing a series of power profiles utilizing different low-power design techniques. Collating the results from said power profiles, a comparison will be undertaken to explore the effect the choice of toolchains, and choice of device will have on the FPGA's ability to compete with microcontrollers.

The team is beginning this project without having previous work to consider within our own designs. We are however leveraging a previous team's toolchain experiences, as provided by Rahul, to select the open-source toolchain that supports the Upduino board, and UltraPlus FPGA we will be using.

Research

Open Source Projects

To simplify the implementation of the project and to avoid reinventing the wheel, Open-source Verilog IPs can be utilized in our design as informed by Rahul. This would also simplify the comparisons between the open source and Lattice toolchain, allowing for the reduction of influences on the final power results. Opencores.com has been recommended by our industry sponsor as a suitable place to explore and obtain said IPs.

Patents, Papers, White Papers, Articles, Conference Proceedings

-The IEEE paper, Development of an FPGA based low power message displaying system using scanning

technique, is an exploration conducted by Ali et al. on developing techniques for the reduction of power usage in FPGAs that interface with a screen. While the system explored in the paper is more complex than the one we intend to design, it provides inspiration on potential avenues of power reduction, and could inspire the methods we utilize in developing and tracking our power profiles and their results.

-Pong game on FPGA with CRT or LCD display and push button controls, by Szabó et al, provides a useful example on where this project can take us. It provided useful information on how we could implement the push buttons for our device such that it can interface correctly with an LCD. It also provides a suitable benchmark on our progress as we proceed to stages 2 and 3 for the project, ie displaying images and text, and would provide a suitable foundation for a fourth stage should we be able to finish with time to spare.

-SystemVerilog based design and implementation of LCD Controller IP Core, by Chandran O et al., provides a suitable source for examining the considerations taken in the design of an LCD controller IP. As it is written in SystemVerilog instead of Verilog, it allows us the opportunity to enhance our understanding of FPGAs by informing us of useful techniques while leaving the burden of implementation on our team. Since the IP being designed is for an LCD controller, this will be useful in determining and visualizing our own mechanism for LCD Interfacing.

3. Product Design Specification

Product Overview

Our goal for this project is to develop an FPGA-based system that controls an SPI-based LCD screen, utilizing an open source toolchain and incorporating at least 3 kinds of low power design techniques.

A key aspect of the project involves comparing the open source toolchain with Lattice Radiant toolchain, focusing on their ease of use , workflow efficiency, and the quality of synthesis results measured by power consumption analysis. For every phase of product development, power consumption data will be systematically collected and analyzed, with visualized results in the form of charts and graphs.

The final product will serve as a demonstration tool for Lattice Semiconductor, emphasizing the advantages of FPGA solution over traditional microcontrollers in terms of cost, efficiency and power consumption. The system integrates two buttons for user interaction, enabling users to adjust display colors and brightness. The system will be housed in a 3D-printed enclosure and the whole system should be battery powered.

Throughout the development process, the system will be divided into three major stages, with progress and results determining the final product phase achieved within this semester. The first stage will focus on implementing basic functionality, allowing users to control the screen's brightness and color using two buttons. The second stage will expand the system's capabilities to display meaningful text on the screen. Finally, the third stage will enable the screen to render images, with potential extensions to connect to sensors, such as a temperature sensor for monitoring environmental conditions.

Stakeholders

1. Industry sponsor : Rahul Koche

2. Faculty Advisor : Roy Kravitz
3. User : Lattice Semiconductor
4. Project team: Nathaniel Fraly, Riley Cox, Mohammad Alshaiji, Xiang Li, Haoyang Han

Requirements

The main priority of this project is to be able to utilize off-shelf products provided by Lattice FPGA to create a proof of concept for an FPGA Color-mixing screen. This can be used in conjunction with an LCD screen utilizing SPI to display color/pattern changes on the press of a button, and contrast/brightness adjustments with the press of another. The proposed boards to be used are the Lattice TinyVision or UPduino boards using either the Crosslink or Ultraplus family of FPGAs. The whole process should utilize an open-source toolchain and GitHub resources to generate the FPGA bitstream.

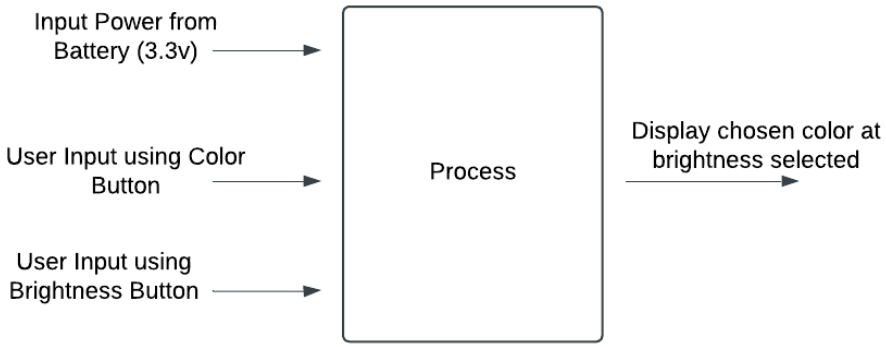
1. Must control an SPI-based LCD screen to adjust brightness and color via two buttons.
2. Must use an Open Source FPGA toolchain to develop FPGA bitstream.
3. Must be encased in a 3D-printed enclosure.
4. Must use at least 3 kinds of low-power design techniques.
5. Must compare the ease of use and efficiency between the Open Source toolchain and the Lattice Radiant toolchain.
6. Must collect and visualize power consumption data and compare the results of the Open Source toolchain, Lattice Radiant toolchain, and Microcontroller-based system.
7. Must have a working final product
8. Should be powered by a battery.
9. Should be able to show meaningful text on the screen.
10. May be able to show images on the display.
11. May be able to act as a debugging tool to show the debug messages on screen.
12. May be able to interact with other sensors.

Specifications

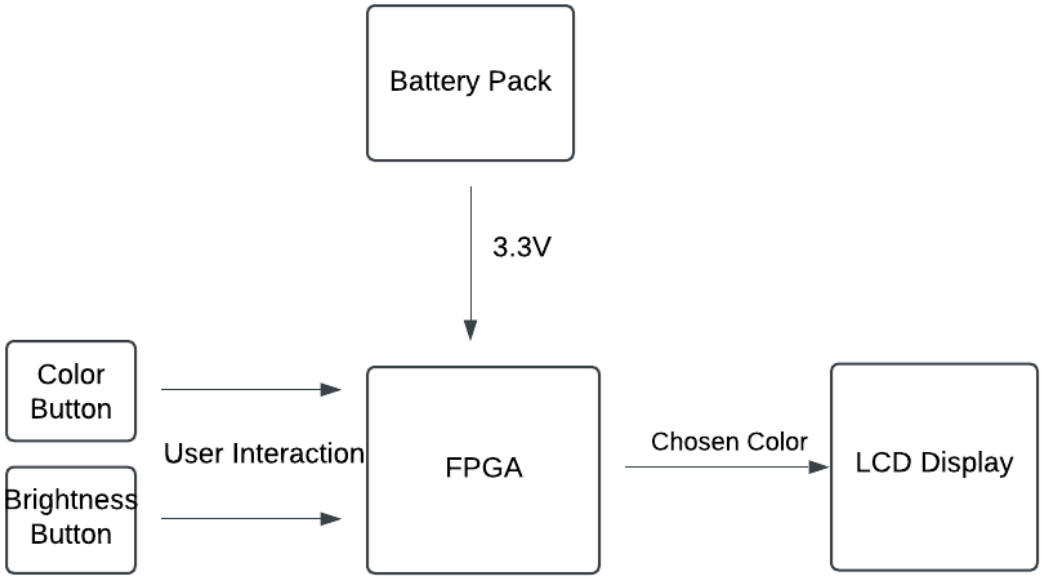
1. Must use the provided UPduino v3.0 board featuring the Lattice iCE40 UltraPlus.
2. Must use Open Source FPGA toolchain to build Lattice FPGA bitstream.
3. Must use the same source code for Open-Source FPGA toolchain and Lattice Radiant toolchain.
4. Must use Adafruit ST7735R 1.8” Color TFT LCD display with MicroSD card Breakout.
5. Must use open-source documentation standards.
6. Should write code in Verilog.

Initial product design

- Hardware architecture
    - **L0 diagram for phase 1:**



- **L1 Diagram for phase 1**:



- User interface/experience
  - At the end of phase one, the user will be able to cycle through a selection of 10-15 colors on the provided adafruit display, with the ability to change color and brightness through the requisite colors
  - At the end of phase two, this functionality will be expanded so that text can be displayed and read from the screen. This is where the main functionality of the device can be showcased, as it would allow for the quick and easy display of error messages, potentially allowing the user to use the device as a debugging tool.
  - At the end of phase three, the user will be able to display more complex visuals on the screen, allowing the user to utilize the device as a viable tester/debugger
- Other considerations
  - At least 3 kinds of low power techniques should be used
  - Collect and analyze power consumption data

Verification plans

Here is a proposed Verification test for “must” requirements:

- For the functioning of the buttons, the brightness button should increase the brightness by an increment on each button press. When the maximum brightness level is reached, subsequent presses of the button should decrease the brightness. The color button should cycle through all color selections available, and when those are done, should cycle back to the first color selection.
- For the 3d printed enclosure, drop and vibration tests can be conducted in order to ensure that the enclosure is able to appropriately handle real world conditions.
- The collation of the power profiles will act as a means to ensure that each power profile utilizes at least three techniques. In order to confirm this, each technique will be initially implemented on its own to compare against the expected result from three implementations at a time.
- To confirm the viability of the toolchain analysis, special attention will be given in ensuring that the code that is fed through both tools is the same, containing the same IPs. The testing of the buttons and device reliability must be conducted exactly the same for both. This is accomplished through the use of a specific written out procedure to ensure the proper accounting of steps.
- The accuracy of resulting power consumption visualizations must be confirmed. This will be done in accordance with the specifications determined by the mechanism of data collection. It would also be prudent to have a base configuration without any power-reduction techniques to act as a control for the comparisons to be made.
- The lifetime and operation for the battery should be tested. This can be done by having multiple packs fully charged and having them drain in the device while under load. An average lifetime can then be determined.

Risks

As things currently stand, our biggest risk is in our ability to properly initialize an open-source toolchain. From our understanding, while Yosys does not directly support the Crosslink and Ultraplus family of FPGAs, we have been informed by Rahul that a previous team has been able to use it with the Crosslink chip. So, our main struggle lies in our ability to finalize our full final chain, as Yosys lacks some functions, such as linters, that would necessitate compensation with other tools.

Another issue is the large scope of the project, as the product is intended to be polished enough to be showcased by the industry sponsor in trade shows. To alleviate the possible issues this can cause, the full project has been divided into three stages. These stages act as checkpoints we can fall back to if our progress on a particular stage doesn't align with our expectations. As such, the product should always be transferable to future teams should the work not be completed.

Deliverables

1. A fully working SPI LCD display which uses FPGA for color-changing, two buttons for functionality and all inside a 3-D printed shell.
2. RTL source code, ideally in Verilog
3. Associated apps or software for LCD display controlling
4. Report for comparison between Open Source toolchain and Lattice Radiant toolchain
5. Reports with statistics and graphs on power consumption of different solutions
6. User Manual
7. Bill of Materials
8. Project proposal
9. Weekly Progress Reports
10. Final report
11. ECE Capstone Poster Session Poster
12. Documentations and source code stored in Github repository

4. Project Management Plan

Timeline, with milestones

Refer to capstone.pdf on the github repository in order to examine a Gantt Chart depicting the timeline of the project from Phase 1-Phase3.

Budget and Resources

Our project operates within a budget constraint set by the industry sponsor, aiming for a final product cost of no more than $60. Key components of the project include the UPduino board with iCE40 UltraPlus FPGA and the Adafruit 1.8" Color TFT LCD display for 19.95$. To stay within the remaining $7, we will carefully manage the costs of additional components such as buttons, wires, batteries, and the enclosure.

We are provided 2 sets of Adafruit LCD display with UPduino Board by our industry sponsor, and we'll use our own computer resources for program development.

Our work will primarily take place in the PSU Engineering Building (EB) and University Pointe (UP) for meetings and development. Additionally, the Electronics Prototyping Lab (EPL) will be a resource for soldering, power consumption testing, and 3D printing.

Intellectual Property Discussion

All IP is owned by the industry sponsor and is to be utilized and licensed as they see fit, within the bounds of any open-source components being used.

Team

- Nathaniel Fraly
  - Skills: (System) Verilog, leadership, adaptability
  - Responsibilities: Team leader.
    Final decision-making, guiding the team, communicating with the industry sponsor and faculty advisor, and contributing to Verilog development.
- Mohammad Alshaiji
  - Skills: Documentation, Verilog, debug, soldering, research
  - Responsibilities: Doing documentation, contributing to Verilog development, and supporting debugging tasks.
- Riley Cox
  - Skills: (System)Verilog, flexibility, research
  - Responsibilities: Conducting research, supporting design and synthesis improvements, managing GitHub Repo and contributing to Verilog development.
- Xiang Li
  - Skills: (System)Verilog, debug
  - Responsibilities: Managing project documentation, and contributing to Verilog development.
- Haoyang Han
  - Skills: Low power reqs, physical layout, soldering
  - Responsibilities: Handling low-power requirements, physical layout, 3D printing, and contributing to Verilog development.

Development Tools and Process

Our development process will follow a spiral methodology, emphasizing iterative cycles of planning, risk assessment, development, and evaluation. We will use this methodology to address uncertainties and refine our project incrementally at every stage. Each cycle will focus on a specific set of objectives to ensure steady progress toward the goal. We will develop prototypes and perform testing during each iteration, followed by evaluations with the team, industry sponsor, and faculty advisor to gather feedback and refine our approach.

Each spiral cycle will include planning objectives, prototyping, and risk assessments. Team members will work on individual branches in GitHub, contributing to development while ensuring quality through a pull request workflow.

Collaboration will be supported by Discord for in-team communication and virtual meetings. GitHub will be our central platform for storing code, documents, and version control. Google Drive will serve as the repository for shared documents, and professional communications with stakeholders will be conducted via email, with Zoom for virtual meetings.

Technically, we will use (System)Verilog as the primary programming language for FPGA design and hardware description. Our FPGA toolchain includes an Open Source toolchain using Yosys and the Lattice Radiant toolchain.