Introduction of Real-World Signals and Systems into ECpE DSP Laboratory Curriculum

Design Document

Team Number: 14

Client: Matt Post

Advisers: Matt Post

Team Members/Roles:
Brady Anderson - Embedded Systems Engineer
Sam Burnett - Hardware Design Engineer
Mitchell Hoppe - Software Engineer
Max Kiley - Manager of Information
Emily LaGrant - Testing and Integration Engineer
Isaac Rex - Hardware Design Engineer

Team Email: sddec20-14@iastate.edu
Team Website: sddec20-14.sd.ece.iastate.edu

Revised: 4/25/20 -- Version 3

Executive Summary

Development Standards & Practices Used

Standard industry convention was used in all hardware and software designs. These include:

- Best practice followed for schematic
  - Use standard signal flow model
  - Separate high level design blocks in to multiple sheets
  - Use standard practice symbols and circuit block drawings
- Best practice followed for PCB layout
  - Keep high impedance traces short
  - Component layout for best routing scheme
  - Keep layout intuitive for end user
- Best practice followed for Power Delivery
  - High efficiency switching regulators
  - Low-noise shielded components
  - Ground stitching, via fencing
- Best practice followed for software
  - Self-documenting code
  - Consistent variable naming conventions according to Python, C, and VHDL standards
  - Atomic Git commits whenever possible
- Standard format used for lab and manual writeups
  - All labs and manuals written in LaTeX
  - 12pt base font
  - Computer Modern Serif font

Summary of Requirements

- Create a platform (CyDAQ) to supplement hardware portion of labs in signals, systems, and communication classes
  - Include a general purpose platform, expandable to many classes (including both lab activities and lecture demos)
- Develop labs for that utilize CyDAQ hardware to connect concepts to real world application for EE 224 and EE 324
  - Test labs to ensure that each lab is completable by an average student in 2 to 4 hours
- Create a set of final projects for each class
- Implement design elements (filter design, controller design, etc.) in labs

Applicable Courses from Iowa State University Curriculum

Electrical Engineering:

EE 201	*Electrical Circuits I*	*EE 422*	*Communication Systems II*
EE 230	*Electrical Circuits II*	*EE 423*	*Communication Systems Lab*
EE 285	*C Programming for EE*	*EE 475*	*Control Systems I*
EE 224	*Signals and Systems I*	*EE 476*	*Control Systems II*
EE 324	*Signals and Systems II*	*EE 333*	*Electronic System Design*
EE 321	*Communication Systems I*		

Computer Engineering:

CPRE 281	*Digital Logic*	*CPRE 381*	*Computer Architecture & Assembly*
CPRE 288	*Embedded Systems I*	*CPRE 488*	*Embedded Systems Design*

Computer Science:

COMS 227	*Object Oriented Programming*	*COMS 319*	*Construction of User Interface*
COMS 309	*Software Development Practices*	*COMS 311*	*Introduction to Algorithms*

English:

ENGL 314	*Technical Communication*

New Skills/Knowledge acquired that was not taught in courses

Technical Knowledge/Skills:

- Hardware design experience
- Hardware manufacturing (assembling, soldering)
- Integrating various aspects of a project into a final product
- Testing and safety procedures for hands-on experiments
- Lab writing

Non-technical Knowledge/Skills:

- Organization and communication skills
- Creative design
- Empathy and emotional intelligence
- Self reflection
- Ethical Leadership Skills
- Cultivating Strategic Communities
- Integrating E-business Solutions
- Orchestrating Leveraged Infrastructures
- Bleeding-Edge Quality Vectors
- Creating Seamless Initiatives
- Delivering One-to-One Sprints
- Seamlessly Targeting Turnkey Metrics

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Acknowledgement

Our team would like to thank the Electronics and Technology Group (ETG) for assistance with equipment, technical advice, and financial aid. We would also like to thank ECpE faculty members Dr. Andrew Bolstad, Dr. Julie Dickerson, and Dr. Shawana Tabassum for their technical support. Without their contributions, our project would not have been possible.

1.2 Problem and Project Statement

Conceptualizing the signals and systems curriculum at Iowa State University can be as frustrating as it can be rewarding for students. In lecture, professors work hard to help students wrap their minds around this difficult curriculum; however, in the lab, the curriculum has fallen behind and currently lacks both the conceptual support for lecture and the real-world application that helps prepare Iowa State students for the professional world. In order to revitalize the signals, systems, and communication classes at Iowa State, a redesign of the signal processing laboratory curriculum is needed.

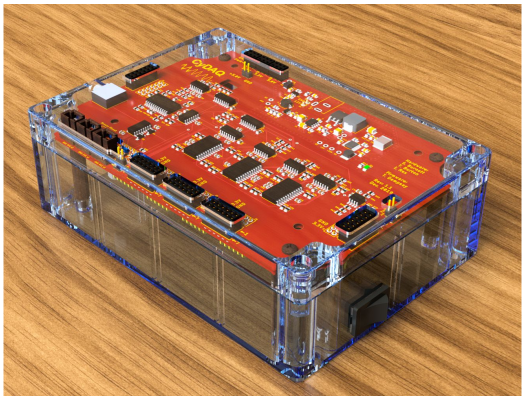


Figure 1: CyDAQ board

At the heart of this redesign is the CyDAQ. CyDAQ, shown in Figure 1, is a hardware teaching platform designed for laboratory curriculum. This board contains onboard analog filters, signal acquisition hardware, and a fully featured powerful embedded computing platform called the Zybo Z7. The Zybo Z7 enables expansion of the CyDAQ base platform by adding external special purpose hardware modules.

CyDAQ has the potential to solve a wide variety of issues found in Iowa State's signal processing course, including but not limited to:

Problem	Solution
● Disconnect between lecture material and real world application	● Show real world application of concepts through hardware experiments
● Design skills and techniques not emphasized	● Introduce design projects that allow the student to learn design process
● Very difficult concepts to internalize and solidify knowledge for future use in class and industry	● Create intuitive sense of concepts by relating them to physical phenomena
● Current labs are pure simulation/Matlab without hardware or real-world connections	● Use CyDAQ and Matlab together to show how these tools can be used in the analysis and design process

CyDAQ is a powerful learning tool that not only has the ability to revitalize the signals, systems, and communication curriculum but also has the potential to spread across the entirety of ECpE curriculum, both inside lecture and labs. By the end of this project, our team hopes to accomplish the following:

- Design a minimum of 4 labs for EE 224 to be implemented in the Fall 2020 semester
- Design a minimum of 4 labs for EE 324 to be implemented in the Spring 2021 semester
- Design a minimum of 2 projects for each EE 224 and EE 324
- Develop a student friendly GUI for CyDAQ platform
- Design and deploy several hardware extensions for CyDAQ, as needed for the designed labs
- Develop lab documents emphasizing real-world analysis and design processes
- Design CyDAQ as a complete learning platform that can be expanded to more classes and labs in the future

1.3 Operational Environment

The CyDAQ will be operating in a standard lab environment, and is expected to be handled by dozens of people each day. In this environment, every-day wear and tear is expected as well as the breaking of various parts. Although food and drink are not allowed in the lab environment, CyDAQ's exposure to food and drink is expected. To counteract these effects, the CyDAQ is designed to be placed in a transparent blue polycarbonate box, along with the FPGA. This transparent casing ensures that damages are kept to a minimum while also allowing students to see the internal components.

1.4 Requirements

Functional Requirements:

Hardware:

- Inputs for a variety of signal sources:
    - General purpose analog differential input
    - 3.5mm TRS auxiliary input jack
    - Special purpose sensors (provided by the team or the ETG)
- Outputs for signal sampling, measurement, and hardware interfacing:
    - General purpose output for signal measurement
    - Stream samples to computer for processing and storage
    - General purpose analog output for interfacing with external hardware
    - General purpose power supply rail for powering external hardware
- On board signal processing
    - Built-in analog filters

    - Ability to implement digital processing of signals
- Multiple input channels for simultaneous analog and digital processing
- Provides interface for expanding the core system to suite future needs

Software:

- The firmware of the FPGA must be able to configure and sample the CyDaq board to meet the requirements of current or potential lab applications with an acceptable amount of stability.
- The UI of the board will allow the user to configure and sample the board using the different I/O and filters available on the board.
    - The UI must not allow users to configure the board in ways that are mathematically impossible or unreasonable.
    - The UI must have a GUI to make it easier for lab students to configure the board.
    - The UI must allow for developer debugging, having functions of configuration and access not available to the GUI

Economic Requirements:

- The final, assembled CyDAQ should cost less than $350

Lab Manual Requirements:

- Labs should have hardware elements and a connection to a real world application
- Labs should be completable by an average student in 2-4 hours
- Final design project should be completable by an average student in 6-8 hours

1.5 Intended Users and Uses

The intended users for all products of this project are electrical and computer engineering students. CyDAQ use will begin in the EE 224 lab during week 10 and continue for the remainder of the Fall 2020 semester. The planned topics and general uses for the CyDAQ in lab curriculum can be found in *Table 1*. This lab curriculum is intended to help EE 224 students understand the learning objectives found in the course's syllabus.

Table 1: Fall 2020 EE 224 CyDAQ lab schedule

Week*	EE 224 Lab Topic
10	CyDAQ basics, signal acquisition, time and frequency analysis
11	Digital filtering of a noisy signal
13	Aliasing
14 & 15	Two week project to filter a signal and provide actionable information

*Weeks not included have non CyDAQ labs

By Spring 2021, integration of labs for EE 324 will begin. These labs will be functionally the same as those for EE 224, but covering EE 324 topics. The content of the labs will cover the learning objectives for EE 324: analysis, design, and implementation of Laplace and Z transforms, signal filters, and feedback system.

1.6 Assumptions and Limitations

Assumptions:

- The CyDAQ will fit in a polycarbonate box
- The design team will create 4+ lab activities for both EE 224, EE 324, and potentially additional classes
- The CyDAQ will be used by students in a lab environment for educational purposes.
- The CyDAQ will always be available for students to access in the signals and systems labs

Limitations

- The final assembled CyDAQ shall cost less than $350
- The CyDaq will have a minimum sample interval of 0.0000010 seconds per sample
- All CyDAQ extension modules necessary for *all* labs must fit inside the lab's cabinet storage space
- Input voltage range is limited to $\pm 5V$
- DAC output voltage range is limited to $\pm 5V$
- Filter bandwidth is $1\ kHz$ to $40\ kHz$

1.7 Expected End Product and Deliverables

There are three primary categories of deliverables for this project: hardware interface, software interface, and lab manuals.

The hardware interface consists of the main CyDAQ platform and "PMOD" hardware expansions as required by individual labs. The main CyDAQ platform is the core of the project, offering students a hardware interface for studying signal processing and control concepts. It is a complete, self-contained unit intended to be stationed at each lab bench: one per computer. It contains the main CyDAQ board with six analog filters, a full hardware interface as detailed in section 1.4 and, a Zybo Z7 embedded system platform for digital control and processing. The initial, functional revision will be due by 3/23/2020. A second revision of the main CyDAQ platform will be due by 12/2020.

Some labs may require additional hardware (e.g., an FM radio receiver, a microphone/speaker, external push-buttons, etc.). These will be delivered in the form of a "PMOD" expansion module. These modules are boards designed to connect to CyDAQ's expansion ports to increase the scalability of the platform. They will be designed alongside their appropriate lab and can be used for future expansion of the platform. The PMODs will be due on a per-lab basis–see the lab section below for due dates.

The software interface will be a front-end GUI and back-end command line used to set up and communicate with CyDAQ. The GUI will be an intuitive interface written in Python for quickly setting up the CyDAQ and transceiving data. The command line will be a lower level interface for developing and scripting of CyDAQ and Zybo. This interface for the CyDAQ will be provided for labs in the form of a .EXE file, such that no dependencies will have to be installed on the lab computers.

The lab manuals will be a full description of a lab experiment for EE 224 and EE 324. They will be written to fulfill the learning objectives of the respective course and provide the student with an intuitive connection between the theory and the application. There will be a minimum of four labs and two projects per class. The lab descriptions will be accompanied by a full solution set per lab.

2. Specifications and Analysis

2.1 Proposed Approach

FPGA Integration using FreeRTOS real-time operating system to handle signal routing, sampling, digital signal processing, and digital to analog conversion tasks simultaneously.

Peripheral module development not limited to:

- Additional digital and analog inputs/outputs with signal conditioning and level shifting
- Radiofrequency couplers, baluns, and antenna interfaces
- Tactile user input and control interfaces
- Signal visualization, indication devices
- Specialized sensor interfaces

Hardware improvements to include on-board digital to analog converter, additional configurable power delivery, and programmable light emitting diodes.

Lab resource development in concert with relevant faculty and in support of established curricular learning objectives. Emphasis will be placed on hands-on intuition of signal acquisition, processing and feedback using the Cydaq as a facilitative device.

2.2 Design Analysis

Current Design Characteristics

Hardware:

- +5V, -5V, +3.3V and +1.8V power delivery
- I2C protocol based filter block configuration
- 100kHz stable passthrough bandwidth
- 40kHz programmable filter bandwidth
- 12bit analog to digital conversion at 1Ms/s
- Limited to single channel input
- 6 programmable filter banks
- 1 channel passthrough

Firmware:

- Bare-metal C application on Zybo PS
- Simple command string formatting with little error detection
- Single-threaded operation, data acquisition and transmission must be done sequentially instead of in parallel
- Data sampling in Zybo PL fabric via Xilinx XADC IP core

Software:

- Rudimentary graphical interface with limited functionality
- Single-threaded tasks for sampling the XADC
- MATLAB App Designer for the graphical components.
- JSON for configuration and data storage

2.3 Development Process

A combination of the waterfall and agile development processes will be used to coordinate the team's progress. High-level development will follow the waterfall process, in that CyDAQ system requirements will be determined early on before the implementation and testing phases begin. This is primarily because faculty will be integrating CyDAQ and it's associated labs into their courses, so the feature set of CyDAQ must be determined early to give them sufficient integration time.

Day-to-day development will follow the agile process model; this will enable team members to independently work on tasks in their focus area, while ensuring consistent progress is achieved each week. A Trello board will be used to coordinate team members and to keep tasks flowing through the development pipeline.

2.4 Conceptual Sketch

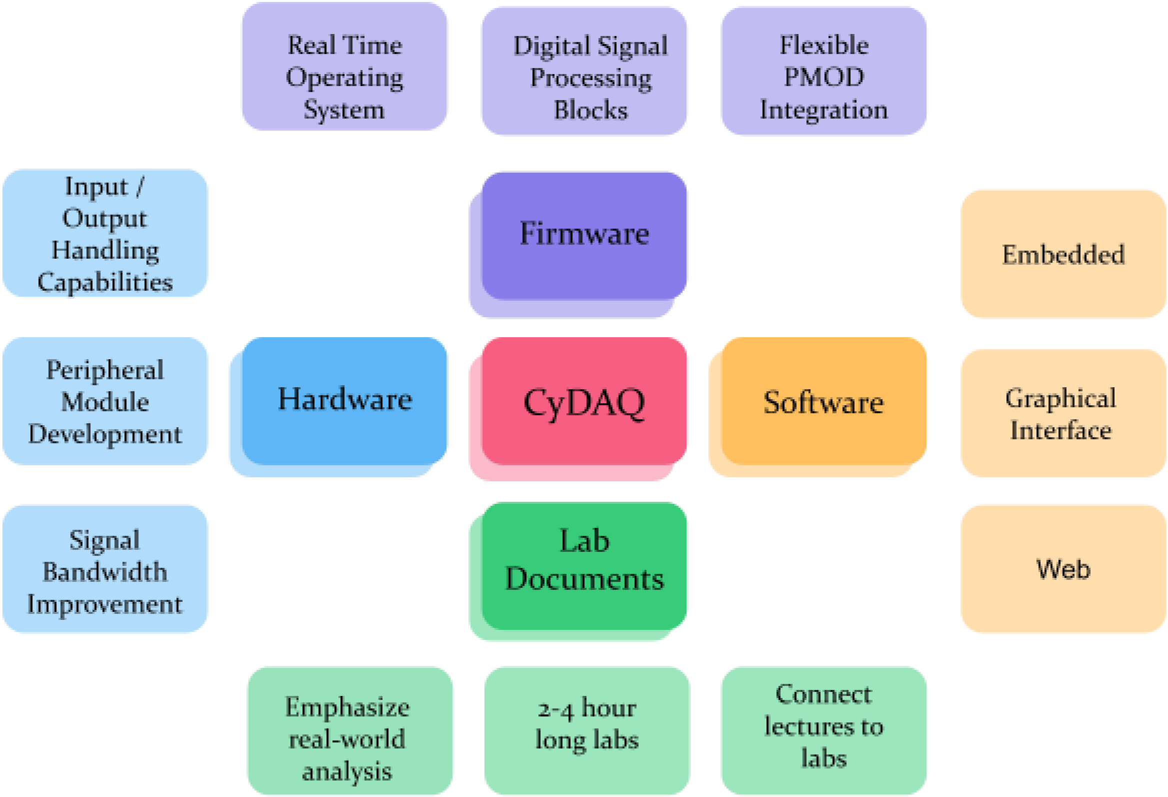
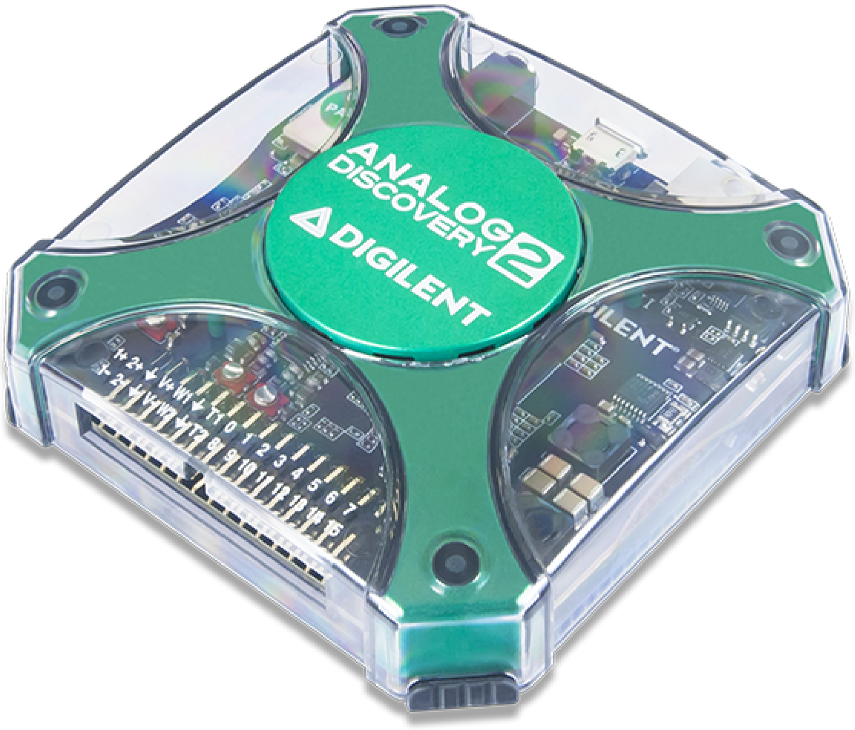

Real Time Operating System
Digital Signal Processing Blocks
Flexible PMOD Integration
Input / Output Handling Capabilities
Firmware
Embedded
Peripheral Module Development
Hardware
CyDAQ
Software
Graphical Interface
Signal Bandwidth Improvement
Lab Documents
Web
Emphasize real-world analysis
2-4 hour long labs
Connect lectures to labs

Figure 2: Conceptual Sketch

3. Statement of Work

3.1 Previous Work And Literature

Data acquisition devices have existed for a long time, but few are targeted directly at the educational market. The major manufacturers are Digilent and National Instruments. Our project will be loosely benchmarked against NI's Elvis III[1], as the Digilent Analog Discovery , shown in Figure 2, is complementary to CyDAQ and will be used alongside it in labs[2]. More research needs to be done in this area.



The distinguishing features of the CyDAQ platform are: low cost, purpose built for the educational environment, smaller footprint, and labs developed alongside the CyDAQ for optimal integration.

3.2 Technology Considerations

The Zybo Z7 is an affordable FPGA board, but lacks some of the features of more advanced Xilinx boards, such as the expanded I/O of the ZedBoard, or the real-time digital video transcoding of the Zynq UltraScale+ MPSoC chipset.

Using a Real-Time Operating System (RTOS) to schedule data acquisition will expand the capability of the sampling interface, but will introduce significantly more software complexity and overhead than writing a bare-metal application.

The hardware filter signal flow allows for great flexibility of signal acquisition, but does not have any means of outputting a signal other than the direct input passthrough.

3.3 Task Decomposition

All team members have been assigned technical and non-technical roles within the group. This creates a natural delegation of tasks within each of the technical divisions of the project.

Non-technical work is distributed as needed and all members understand the need for collaboration and synergistic workflows.

There are three major tasks that work interdependently. These are:

- GUI development
- FPGA firmware
- Lab Development

While the labs can be developed for our team's new GUI, the labs must also be compatible with previous versions of the GUI in order to reduce interdependence. This also creates a more relaxed timeline as the GUI is an incredibly labor intensive task. While the FPGA firmware is also vital to the final design, it too can wait until after the initial rounds of lab development have taken place as there are previous versions of the firmware available. Once the EE 224 labs have entered the curriculum, our team will be taking the time to evaluate feedback obtained from professors and students.

These tasks and their timeline are also outlined in Table 2, section 4.1.

3.4 Possible Risks And Risk Management

Team members being unknowledgeable for completing a particular task is a possible risk. Not all experience in the required areas is covered in the skillset of the group. The solution to this is group members will learn new skills and technical abilities as the need is encountered.

Adding hardware components to an expensive FPGA platform carries the possibility of damaging its electronic systems. Team members must be knowledgeable about the design of hardware components and their pinouts before attempting to connect them to the Zybo.

These hardware components also carry a risk of delaying the timeline due to the delay of shipped goods caused from the COVID-19 pandemic. Although this risk can be decreased by using reliable sources for hardware components, delays in manufacturing may impact minor milestones.

3.5 Project Proposed Milestones and Evaluation Criteria

The group will begin by focusing on the creation of labs for EE 224 and EE 324 with the potential of also creating labs for a variety of classes within the ECpE Department. In addition, the CyDAQ boards will be distributed to accompany all labs.

Our milestones thus far are to complete multiple labs for EE 224 and begin the design process for EE 324 labs. Updating the GUI and other software interfaces on the CyDAQ is also vital to the project. Once this has been completed, our team will be evaluating feedback obtained from professors and students. In particular, we will be focusing on the duration, difficulty, and organization of the labs for evaluation criteria.

3.6 Project Tracking Procedures

The group will use Trello to track current work items, assign members to specific items, and track completion rates for the work items. New work items are created from consensus from the group during the weekly technical meetings. Code changes will be tracked through git version control. Code reviews and versioning will be handled through Gitlab, where new changes have to be approved by other members of the group before they go out to production.

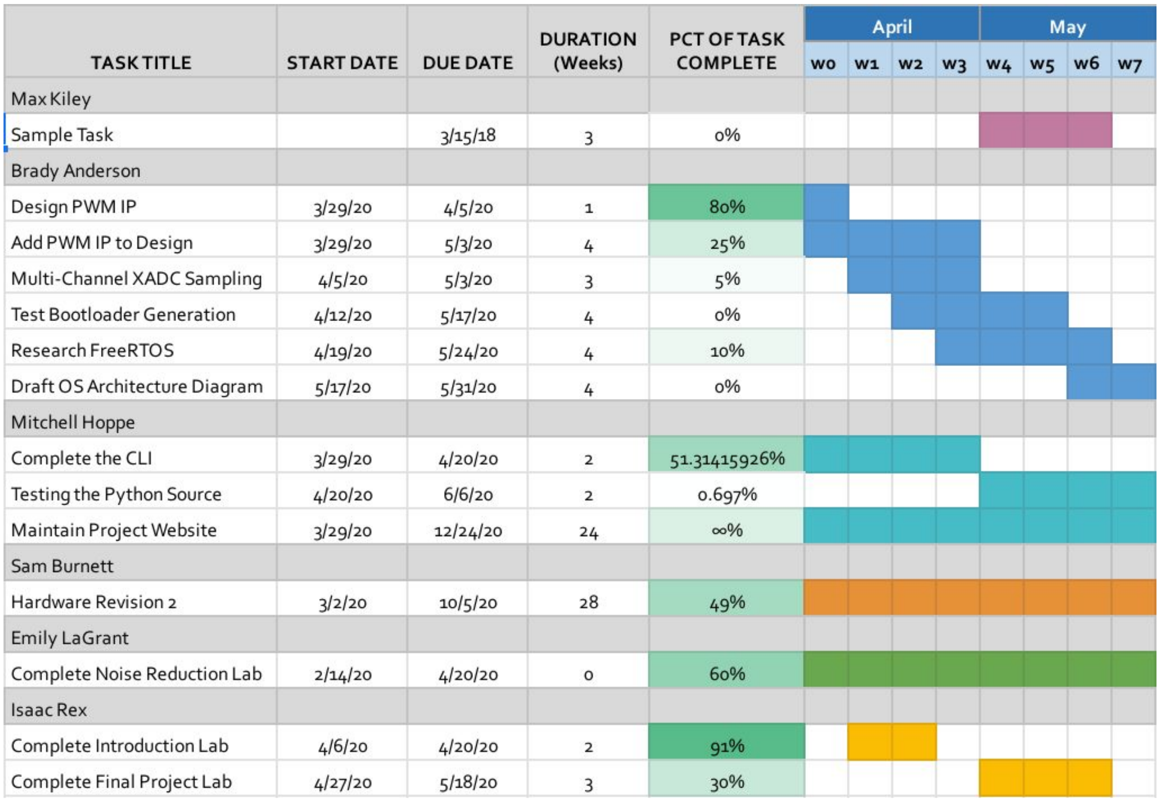
3.7 Expected Results and Validation

The ultimate outcome of this project will be an increase in student's understanding and enjoyment of the signals and system curriculum. This will be measured by performing test implementations in the labs during the Spring and Fall 2020 semesters. To ensure that this goal has been achieved, the professors that administer the labs will relay feedback from students about the labs, their content, and if there was a significant improvement of understanding of the material over the old labs.

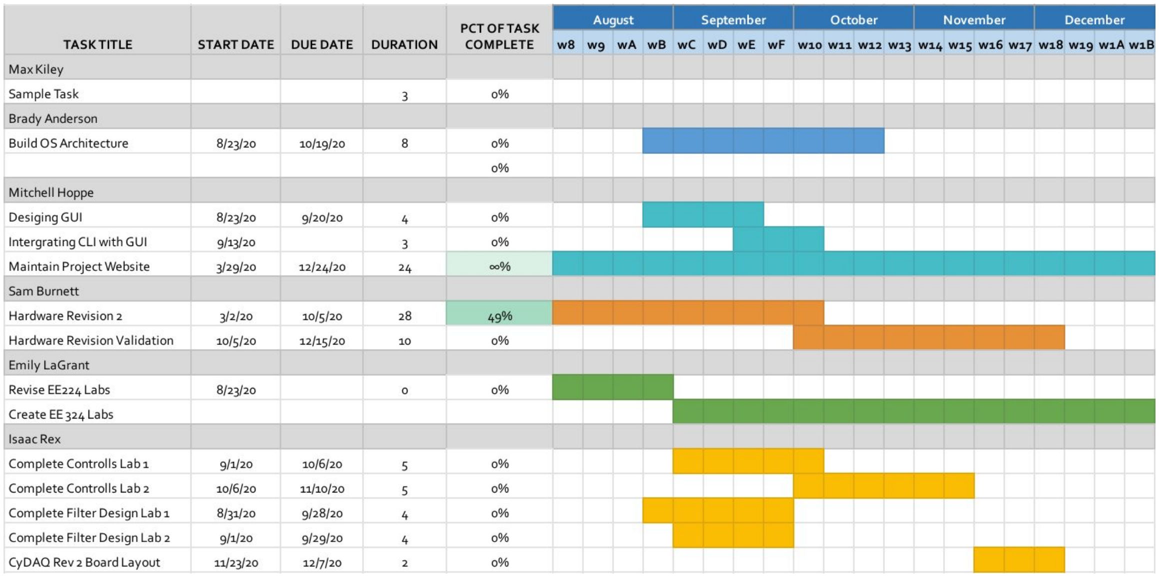
4. Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

Table 2: Project Timeline



TASK TITLE	START DATE	DUE DATE	DURATION (Weeks)	PCT OF TASK COMPLETE	April				May			
					w0	w1	w2	w3	w4	w5	w6	w7
Max Kiley												
Sample Task		3/15/18	3	0%					🟪	🟪	🟪	
Brady Anderson												
Design PWM IP	3/29/20	4/5/20	1	80%	🟦							
Add PWM IP to Design	3/29/20	5/3/20	4	25%	🟦	🟦	🟦	🟦				
Multi-Channel XADC Sampling	4/5/20	5/3/20	3	5%		🟦	🟦	🟦				
Test Bootloader Generation	4/12/20	5/17/20	4	0%			🟦	🟦	🟦	🟦		
Research FreeRTOS	4/19/20	5/24/20	4	10%				🟦	🟦	🟦	🟦	
Draft OS Architecture Diagram	5/17/20	5/31/20	4	0%							🟦	🟦
Mitchell Hoppe												
Complete the CLI	3/29/20	4/20/20	2	51.31415926%	🟦	🟦	🟦	🟦				
Testing the Python Source	4/20/20	6/6/20	2	0.697%					🟦	🟦	🟦	🟦
Maintain Project Website	3/29/20	12/24/20	24	∞%	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦
Sam Burnett												
Hardware Revision 2	3/2/20	10/5/20	28	49%	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧
Emily LaGrant												
Complete Noise Reduction Lab	2/14/20	4/20/20	0	60%	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Isaac Rex												
Complete Introduction Lab	4/6/20	4/20/20	2	91%		🟨	🟨					
Complete Final Project Lab	4/27/20	5/18/20	3	30%					🟨	🟨	🟨	



TASK TITLE	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE	August				September				October				November				December			
					w8	w9	wA	wB	wC	wD	wE	wF	w10	w11	w12	w13	w14	w15	w16	w17	w18	w19	w1A	w1B
Max Kiley																								
Sample Task			3	0%																				
Brady Anderson																								
Build OS Architecture	8/23/20	10/19/20	8	0%				🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦									
				0%																				
Mitchell Hoppe																								
Desiging GUI	8/23/20	9/20/20	4	0%				🟦	🟦	🟦	🟦													
Intergrating CLI with GUI	9/13/20		3	0%							🟦	🟦	🟦											
Maintain Project Website	3/29/20	12/24/20	24	∞%	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦
Sam Burnett																								
Hardware Revision 2	3/2/20	10/5/20	28	49%	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧											
Hardware Revision Validation	10/5/20	12/15/20	10	0%									🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧	🟧			
Emily LaGrant																								
Revise EE224 Labs	8/23/20		0	0%	🟩	🟩	🟩	🟩																
Create EE 324 Labs									🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Isaac Rex																								
Complete Controlls Lab 1	9/1/20	10/6/20	5	0%					🟨	🟨	🟨	🟨	🟨											
Complete Controlls Lab 2	10/6/20	11/10/20	5	0%									🟨	🟨	🟨	🟨	🟨	🟨						
Complete Filter Design Lab 1	8/31/20	9/28/20	4	0%				🟨	🟨	🟨	🟨	🟨												
Complete Filter Design Lab 2	9/1/20	9/29/20	4	0%					🟨	🟨	🟨	🟨												
CyDAQ Rev 2 Board Layout	11/23/20	12/7/20	2	0%															🟨	🟨	🟨			

The majority of the work being done in the group will be focused on writing the labs given the existing hardware and the software. Small revisions to the boards will be created to add functionality and software will be built-out and maintained to support the added hardware functions. All revisions to hardware, software and labs will be developed concurrently as new requirements emerge from the revisions.

The timelines proposed are decided on a Baysian analysis from each respective member. Judging how long each task will take based on previous experience. Timelines are very volatile and subject to change based on emerging technical discoveries, roadblocks hit during the task that will take an investment of resources to solve, and prerequisite work items to complete that were previously undiscovered.

4.2 Feasibility Assessment

By the end of the project, the team should have developed 8 or more labs—4 for EE224 and 4 or more for EE324. Hardware will be distributed to the appropriate lab computers, all labs will be completed and tested for classroom use, and the embedded hardware and software to support such labs will be tested and installed on relevant lab computers. An intuitive GUI will be developed for students to complete what is required in the labs.

Although we anticipate many challenges, our team does not believe that they will cause any major delays to our deliverables. In the upcoming months, it will be difficult to test and meet with our clients, but not undoable. Any meetings will transition from person-to-person to online. Receiving feedback will also be affected as the EE 224 labs have moved online and will appear slightly different than the final CyDAQ version. Once again, this is not a major challenge since the duration, difficulty, and organization of the labs will not significantly change.

4.3 Personnel Effort Requirements

Detailed estimates of personnel requirements and itemized work efforts are detailed in section 4.1 above.

4.4 Other Resource Requirements

Because the majority of CyDAQ hardware development has already been completed by ETG, we require very few non-financial resources. The CyDAQ units are already assembled, so any changes we make will only be PCB revisions or software development.

We may need software licenses (such as MATLAB) to allow students to complete the labs. Since we are designing these labs for professors to use, we will certainly need their feedback and the feedback of students to make sure the labs can be completed and are educational. We will also need access to lab testing equipment like oscilloscopes, function generators, and the Digilent Analog Discovery boards to verify new changes and for students to view the system outputs.

4.5 Financial Requirements

Hardware development support including DAC and PMOD prototyping and PCB revision expenses. ETG employee rates if professors require future lab development after the members of this team have graduated.

5. Testing and Implementation

5.1 Interface Specifications

The FPGA board interfaces with the lab computers through a serial protocol to configure and send/receive data between a PC and the CyDAQ board. To upload firmware and perform embedded testing, the program Vivado allows the team to upload and test new versions of the firmware.

We will be using a custom command line interface (CLI) protocol over UART to communicate with and debug CyDAQ's software. The following table outlines the basic command strings.

Table 3: CyDAQ UART command strings protocol

Hardware interface testing will use special breakout boards designed to interface with the Zynq's PMODS. These will be used for doing development of peripheral hardware or hardware additions for the next revision. Using these PMOD breakouts allows for rapid prototyping of new circuits, and it allows extensible features to be highly scalable. Figure 5 below shows the breakout board used to prototype the DAC that will be added to the next hardware revision.

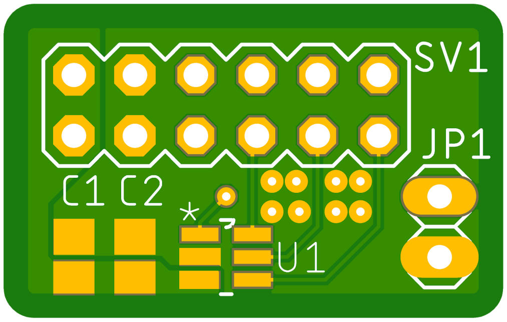


Figure 4: DAC prototyping PMOD

5.2 Hardware and software

Vivado and the Xilinx SDK will be used to produce and flash the new firmware onto the FPGA. Vivado is the Xilinx made software development tool for designing and testing hardware designs on an FPGA.The SDK is the Xilinx provided environment for programming and flashing the microprocessors on the FPGA.

The MATLAB UI will be used for testing to configure the board in different ways to cover the requirements of the labs. The MATLAB UI is the interface that will be presented to students working with CyDAQ. It’s used to set up and configure the CyDAQ for the different modes of operation needed in the labs.

Hardware testing will be done using the Electrical Engineer’s bench top tools: power supply, function generator, oscilloscope, and digital multimeter. These tools allow the subsystems to be tested and debugged independent of the overall system, validating each component at the block level.

5.3 Functional Testing

Lab activity dry-runs will be completed by team members to ensure device functionality and verify completion timeframes. Hardware, firmware, and software stress testing will be completed individually and in an integrated environment, and involve sustained operation at maximum capabilities for an extended period of time.

5.4 Non-Functional Testing

Hardware:

- Channel bandwidth limit of CyDAQ
- Physical durability (drop test)
- Ease of use

Software:

- Data transfer time
- Ease of use

5.5 Process

Vivado's HDL simulation software was used to verify the functions of hardware blocks designed for the FPGA. SDK was used in debug mode to step through the firmware and trace the logical flow that particular command strings create. This was especially important for reconstructing the command string format so that the front-end can utilize the correct data transfer format.

To test lab manuals, we start by having team members who did not write the lab complete the lab and give initial feedback. Then the lab is sent to our advisor and the instructors for the respective class. Lastly, the lab is deployed and feedback is received from students through the course TAs and instructor.

To test the PMOD hardware,controlled test signals and real signal inputs were measured and validated for specified performance parameters. Peripheral devices behaved according to designed specifications.

See Figure 5 below for our test process flow chart.

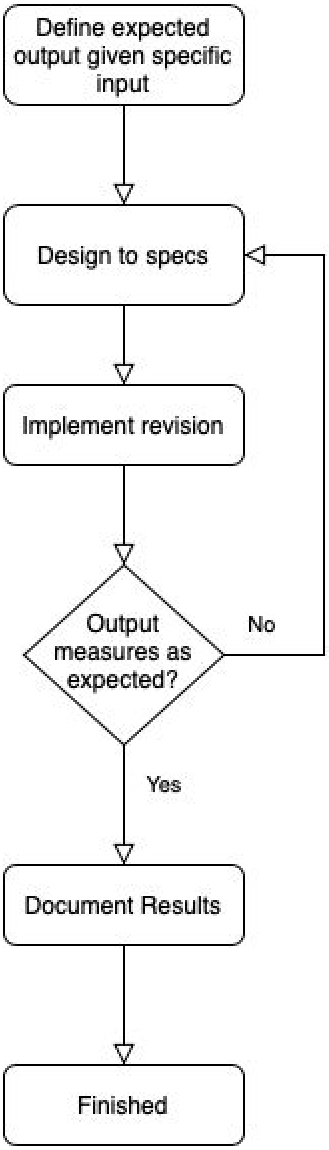

Define expected output given specific input
Design to specs
Implement revision
Output measures as expected?
No
Yes
Document Results
Finished

Figure 5: Test Process Flow Chart

5.6 Results

Running our HDL designs through Vivado's HDL simulator revealed several minor flaws in the initial design approach. Certain operations did not scale their outputs in time correctly, and so when the HDL configuration was changed, these operations did not behave correctly. The purpose of testing is to catch and fix such flaws, however, and after a few revisions the simulations behave as expected.

This semester, we deployed our first labs in EE 224, but have not received detailed feedback from the instructors yet. Our main lab testing phase has not yet commenced, but is scheduled for early Fall 2020. As a result, we have no test lab manual results to report at this time.

The PMOD hardware was tested as described in Figure 5, using a known generated signal and measuring the output with an oscilloscope. All results were exactly as expected, and the modules can now be fully implemented into the final revision.

6. Closing Material

6.1 Conclusion

In order to revitalize the signals and systems curriculum at Iowa State University, a major rework of laboratory curriculum is required. The CyDAQ is at the heart of this rework and can lead to students successfully applying lecture content to real-world applications. This new lab curriculum focuses not only on real-world application but also allows students to learn different design processes as they are given the opportunity to create design projects. CyDAQ has the potential to solve these issues found in the curriculum and more as CyDAQ has the potential to expand its use to classes all across the ECpE department.

6.2 References

[1] National Instruments ELVIS III, https://www.ni.com/en-us/support/model.ni-elvis-iii.html

[2] Digilent Analog Discovery 2, https://store.digilentinc.com/analog-discovery-2-100msps-usb-oscilloscope-logic-analyzer-and-variable-power-supply/

6.3 Appendices

None at this time.