UVU MCS Graduate Paper

BENJAMIN STONEKING, Candidate
FRANK JONES, Advisor

The project explores the process of implementing an embedded real-time digital audio synthesizer. Emphasis is placed on leveraging the inherent strengths of the hardware (timers, hardware interrupts, etc) to produce a reliable musical instrument. The project is a focused exercise in implementing a digital audio processing pipeline with software implementations of common components such as oscillators, pitch control, signal gain attenuation, and filtering and it illustrates the process of implementing a capable, complex and extendable embedded system on a platform with limited processing power. The final product is a fundamental subtractive synthesizer: It responds to input from generic MIDI devices to produce an audio signal. Volume dynamics and timbre control are added to the signal with envelopes and filters. The device has a hardware interface consisting of knobs, encoders, and a screen allowing direct real-time manipulation of the processing. A PC application was implemented to control the system remotely. The system is robust, but many features remain to be implemented. Implementing the expected functionality of electronic musical instruments was challenging. The project code base is close to 2250 lines. Substantial time was spent writing drivers to interface with hardware which often added up to very few lines of code. Processing speed and instruction space is limited and large code bases can quickly outgrow the capabilities of the device requiring efficient and concise code. Substantial effort and time was required to prototype the hardware interface on a breadboard and revise as new components were integrated. More time was required to design and build a permanent hardware fixture on a circuit board.

Additional Key Words and Phrases: audio, synthesis, embedded, microcontroller, real-time, midi, hardware, filters, envelopes, DSP

ACM Reference Format:
Benjamin Stoneking and Frank Jones. 2023. UVU MCS Graduate Paper. 1, 1 (April 2023), 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Begin with a proper thesis statement, which may be several paragraphs. A proper thesis statement answers the following 4 questions:

(1) What problem are you trying to solve?
(2) why is it interesting?
(3) What are your major results?
(4) What are your conclusions?

This is a technical paper, so you need to provide background and evidence for any claims you make about your work and others' work, with citations.

Authors' addresses: Benjamin Stoneking, Candidate, benjamin.stoneking@protonmail.com; Frank Jones, Advisor, frankj@uvu.edu.

XXXX-XXXX/2023/4-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

It is helpful to your advisory committee if you also address these two questions briefly. You will say more about these topics in other sections of the paper.

(1) What did you learn?
(2) How many lines of code is your project?
(3) If not lines of code, what makes your project complex?

The size and scope of your project should be large enough to require 5000 lines of code and take 10-20 pages to document. If you need help with writing, contact the UVU Writing Center.

You may use any editing software you want to produce your paper, but you must follow this template, and the final version you submit must be a PDF. In case you're interested, this template is a customized version of the ACM Sample Manuscript Template. We recommend using either MSWord or LaTeX for writing.

The last paragraph of a good Introduction outlines the sections in the rest of the paper: One sentence per section. Guide the reader, don't surprise them.

2 RELATED WORKS

For each reference in your paper, write a two- or three-sentence paragraph describing how it relates to your work. Think of this as an annotated bibliography covering your literature search in an organized progression that highlights your how your work is similar and different. This is particularly important if your project repeats previous work. At least 15 sources is a reasonable number.

Some authors prefer to include related works as part of the Introduction, because there is overlap. For project defense purposes, it best to have it as a separate section.

3 CONCEPTS

This section is **optional**. Include it if you have ideas that a reader needs to know to understand what you did, but which is too much for either Introduction and does not fit naturally in other sections.

4 SOFTWARE ARCHITECTURE AND IMPLEMENTATION

Describe here the design and implementation of your system. **This section is the heart of your paper.** Include figures that illustrate structure and behavior, especially critical aspects of the design and implementation. You should not detail every aspect of the project, just the most important information. Include:

(1) Meta-Architecture: Choices that influence your design and implementation, but are not those.
(2) Conceptual Architecture Diagram: A one-page "brochure-style" drawing of the concept and vision of your project.
(3) Structure and Behavior diagrams for architecturally significant parts of your system.
(4) screen shots
(5) example inputs/outputs for important scenarios
(6) key aspects of the implementation
(7) factual, objective discussion of implementation successes and challenges

You may decide to discuss Implementation/coding issues in a separate section, but that is up to you.

5 PROJECT RESULTS

The Results section is a technical assessment of your project, keeping in mind that it is foremost a learning experience.

(1) What successes did you have?
(2) What failures?
(3) What was completed?
(4) What was planned but not completed?
(5) Looking back, what early choices were good or bad?
(6) What performance issues or bottlenecks exist in your project?

Important things to keep in mind:

(1) An incomplete project is still a success if you learned from it.
(2) Failure is as important a result as success when doing research. The Wright Brothers did not fly on their first attempt, but they did not give up.

6 CONCLUSION

The conclusion should address the following items:

(1) Summarize and synthesize your learning and effort on the project.
(2) What is your personal view on the success or failure of the project?
(3) What did you learn in creating the project?
(4) What would have been helpful to know before starting the project?
(5) What was your most important results and conclusions?
(6) What are potential next steps for the project?

The rest of the paper needs be factual and objective in tone, but in the conclusion you are free to express your opinions, beliefs and feelings.

7 ACKNOWLEDGEMENT

This section is optional.

8 FIGURES AND CAPTIONS

Remove this section in the finished paper. Figures have to be legible to be useful. LaTeX will automatically number figures. In Word you have to do that by hand.

Captions should tell the user what they are seeing and why. Don't make the reader hunt in the text for basic information.

9 TABLES

Remove this section in the finished paper. Format tables in nice columns with headings. Use only horizontal lines, no vertical lines. LaTeX will automatically number tables. In Word you have to do that by hand.

10 REFERENCES

References should follow ACM format style.

Using BibTeX with LaTeX for preparing and formatting references is strongly recommended because it does the right thing for you. In Word you have to manage references and citations by hand or use a third-party tool.

REFERENCES

A USER'S MANUAL

The User's Manual section contains instructions for installing and running the project. Give sufficient detail that someone other than the candidate could install and use the software if they had prior knowledge of the domain. **This User's Manual is a critical section of the paper and you cannot pass your defense without it.**