Report on the Learnings from Designing a PCB:
Desktop Pomodoro Timer

EE 256 – Board Level Design

Shane Blinkman – December 12th, 2024

Abstract

This report details the process of designing a desktop Pomodoro clock PCB as part of the EE 256 course at Stanford University. The project focused on developing skills in PCB design, troubleshooting, and iterative improvement while navigating real-world challenges. Despite initial setbacks and design pivots, the outcome provided valuable insights into PCB assembly and debugging, laying a foundation for future projects.

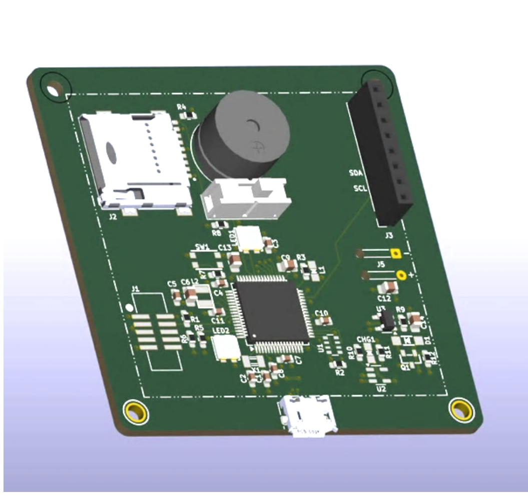
1. Project Introduction and Requirements

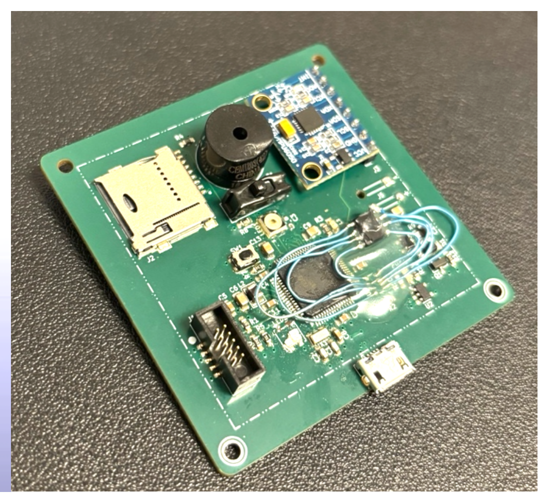
This project focused on developing skills in designing and fabricating a printed circuit board (PCB) with an emphasis on debugging and troubleshooting rather than aesthetics or functionality. The aim was to learn through making mistakes.

The project requirements included using the ATSAMD51J19A-AUT microcontroller with circuitry for power regulation and single-wire debug (SWD) programming. It also required 2MB flash memory, a microSD card interface, a Neopixel LED, red LED and orange LED for event and charging indications, and a micro USB interface for programming, power, and charging. The board had to fit within a 99mm x 99mm size limit, use a maximum of four layers, and incorporate only legged components.

2. Solution

During the project, I pivoted to a new design due to functional limitations in my original idea. Initially, the project was a smart beverage coaster that illuminated colors reflecting the beverage's temperature. Limitations in the temperature sensor made the functionality of the design unfeasible. The project switched to a desktop Pomodoro timer[1] controlled by a gyroscope, allowing users to set, start, and reset the timer by tilting the device in various directions.





[1] A Pomodoro Timer is a time management tool that divides work into focused intervals, typically 25 minutes long, separated by short breaks to enhance productivity and focus.

Image 1: On the left is the 3D rendering of the PCB, and on the right is the fabricated PCB. The 2MB flash component is prominently visible, mounted dead-bug style and connected to the pads using blue 30 AWG wire. A closer look reveals the melted plastic on the active buzzer and slider switch.

The following sections detail the components of both original and final project.

2.1. System Architecture

The block diagram in Appendix A outlines the initial project' system's connections, communication protocols. Due to time constraints, no updated block diagram was created for the updated project.

The updated project featured several subsystems. The power and voltage regulation subsystems were pre-provided, along with the microSD system which communicated with the MCU via SPI. Also, pre-provided, the 2MB flash memory used QSPI communication. A Neopixel LED and active buzzer used single-wire and digital output communication, respectively. The primary sensors were an inertial measurement unit (IMU) communicating via I2C and a three-wire sliding switch.

2.2. Component Selection

Many components were predetermined. The ATSAMD51J19A-AUT microcontroller was selected for its high performance, real-time data processing, and rich peripherals. The microSD interface, 2MB flash, and micro USB were chosen for bulk ordering and structural integrity. The MPU6050 IMU module was used due to availability. The slider and active buzzer were selected for affordability and ease of integration.

2.3. Schematic Design

The schematic design incorporated existing designs and added components as needed. The microcontroller's schematic was based on the Feather M4 board as see in appendix B. The IMU module required only an 8-pin connector to interface. The slider switch and active buzzer were simple, requiring minimal components. The microSD interface included a 4.7K pull-up resistor on the chip select line as specified in the datasheet.

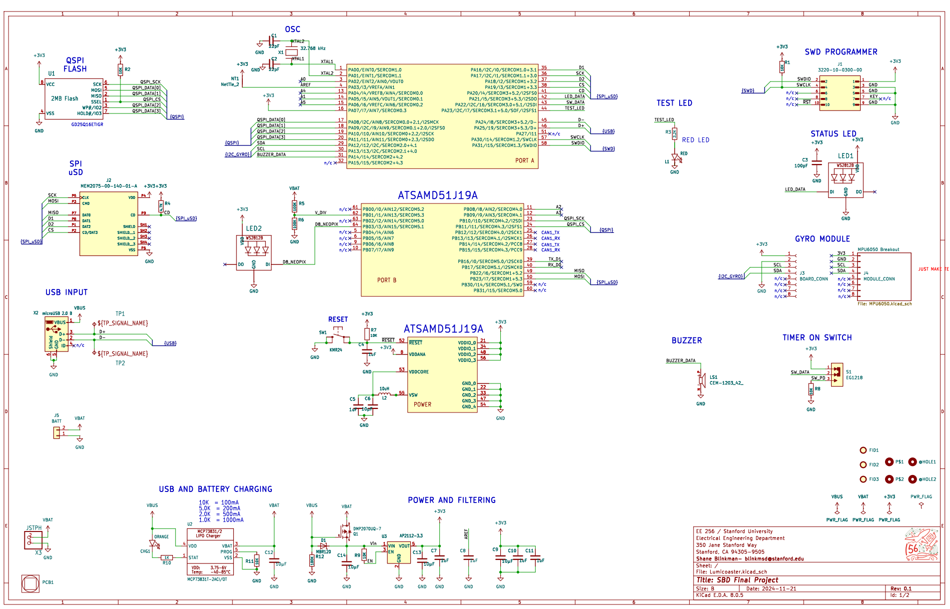


Image 2: The complete schematic of the project. All subsystems described are present.

2.4. PCB Layout

PCB layout was one of the most challenging aspects of the project, requiring attention to trace width, via placement, and component footprints to ensure a functional design. Using KiCad, I utilized class labs and supplier resources for accurate footprints and 3D models. The larger board size helped simplify the process by allowing for easier component spacing, reducing congestion, and making routing more straightforward. This approach also improved manufacturability and minimized potential layout issues.

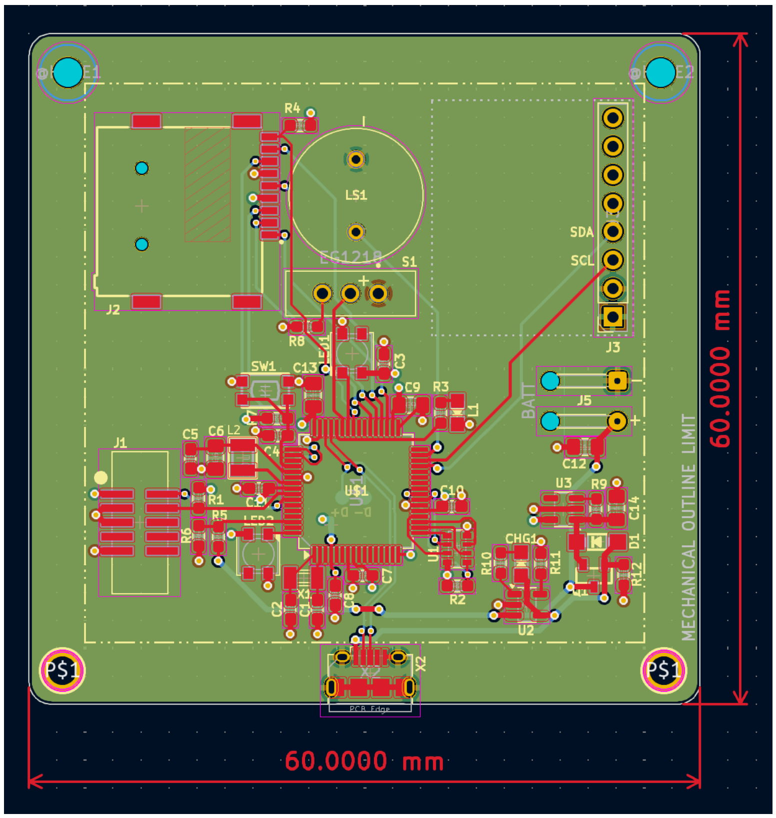


Image 3: The routed PCB layout with the dimensions of the board visible

3. Challenges

The challenges faced during this project highlighted the critical importance of preparation and attention to detail in PCB design. One significant challenge was a mismatched footprint for the 2MB flash memory. This issue required manually connecting the component to its pads using 30 AWG wires, a frustrating and labor-intensive process that could have been avoided with early verification. The error occurred because the imported footprint was assumed to be correct without validating it using a 3D model, which would have revealed the mismatch before fabrication.

Another challenge stemmed from the Neopixel LEDs, where conflicting pin-one indicators in the datasheet and footprint documentation, which can be referenced in appendix C, led to incorrect installation. The misaligned Neopixels disrupted SWD programming due to the high current draw limiting voltage power from the micro USB. Resolving this required isolating power and ground connections with a bench power supply to identify the root cause. During troubleshooting, both Neopixels exploded, necessitating replacement. The process of removing the damaged components with a heat reflow gun inadvertently caused collateral damage, including melted plastic on the adjacent sliding switch and active buzzer, rendering the switch unusable.

4. Learning

This project emphasized the necessity of thorough preparation and iterative validation. Verifying component footprints by rendering 3D models before production emerged as a key lesson, as these steps can prevent costly errors and streamline assembly. The challenges with mismatched footprints and reversed components reinforced the importance of meticulous attention to detail when interpreting datasheets and integrating components.

Through these obstacles, I developed practical debugging and repair skills, such as manually connecting components with high gauge wire and safely soldering delicate parts. These experiences also improved my ability to test circuit continuity, resistance, and capacitance, ensuring accurate functionality and enabling more efficient troubleshooting.

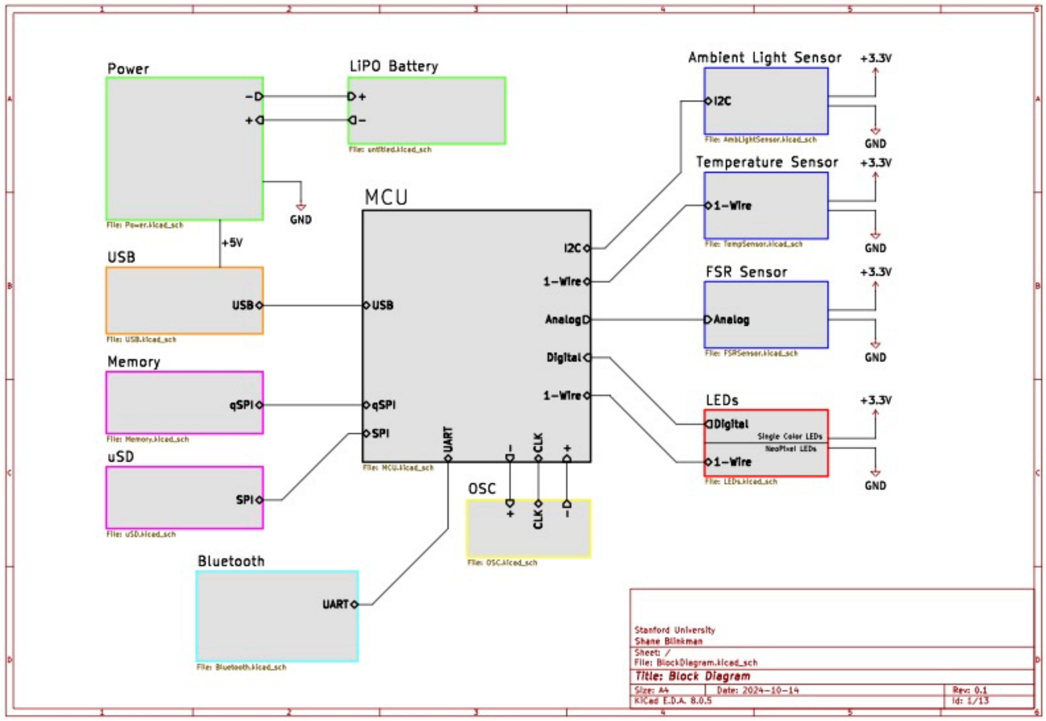
Moreover, the project underscored the importance of adhering to the original scope. Pivoting to a new design late in the development cycle introduced unnecessary complexity and diverted focus from mastering PCB design and layout tools. Maintaining the initial design would have provided a more stable foundation for learning and achieving the project's educational objectives.

5. Conclusion

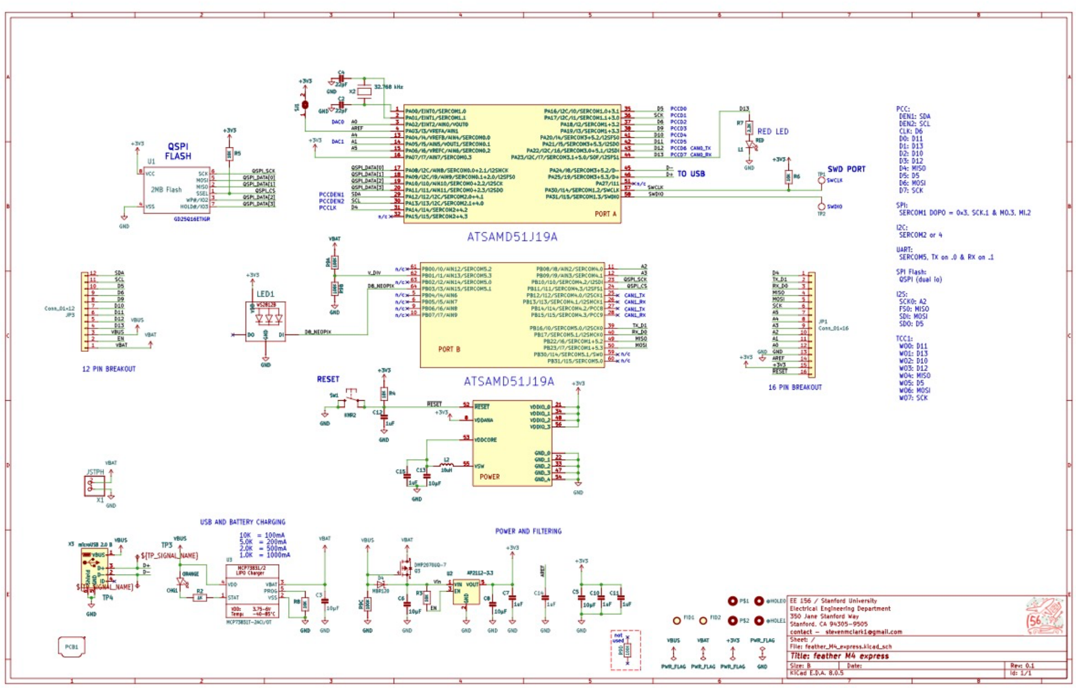
While the project did not achieve full functionality, it provided valuable lessons in PCB design and development. Future projects will benefit from improved planning, iterative testing, and documentation. I feel confident in my debugging skills and understanding of PCB design considerations.

Appendix

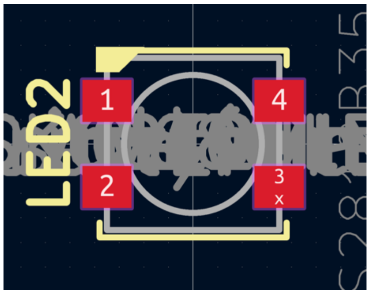
A) Block Diagram

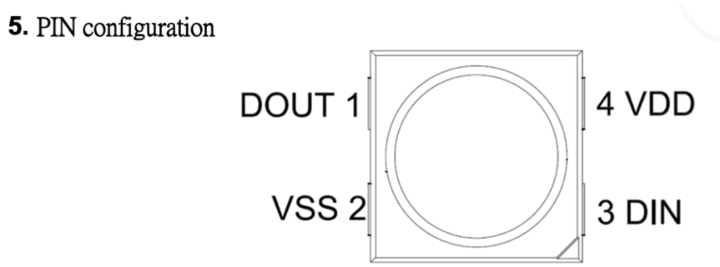


B) Feather M4 board schematic



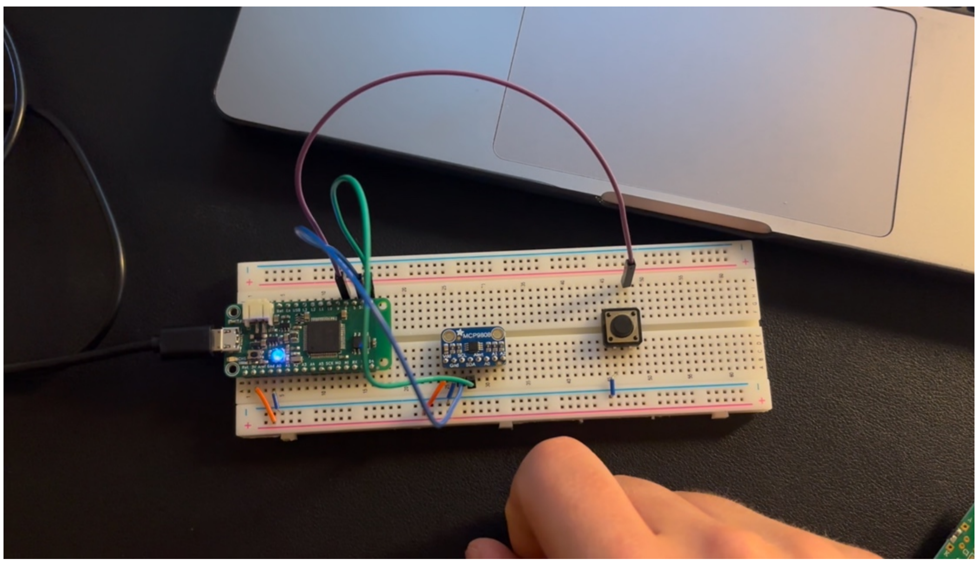
C) Discrepancy between Neopixel footprint and datasheet





As seen in the screenshots, the footprint of the Neopixel has the pin indicator at pin 1, while the [datasheet] has the pin indicator on pin 3. After reviewing more thoroughly, the pin indicators were meant to be matched, which I failed to do.

D) Prototype of smart coaster



E) Prototype of the pomodoro timer