Final Report
Reflow Oven for University Lab

08.16.2021

—

Brandon Hu CS
Micah Rice CE
Mihretab Desta EE
Rohan Mathew EE

Faculty Advisor: Dr. Rick Cordray
Industry Sponsor: Dr. Arnie Berger

Table of Contents

Introduction

Background and Motivation

A reflow oven can be purchased depending on the budget and intentions of the buyer at a price between $50 and $50,000 USD. Common characteristics of high-end ovens include high-quality heating plates and elements, well-designed and implemented software that allows the user to program and save several custom temperature profiles. This last feature allowing for custom profiling can be used to fit different types of PCB requirements. A small PCB can be heated quickly using a large element, allowing for a shorter time period to be programmed in the temperature profile. This can save a great deal of time working through multiple soldering cycles for many different PCBs cumulatively.

At the University of Washington Bothell Labs, there is a use case for this type of high-end functionality as students and faculty work through using the reflow oven to solder many different PCBs. The goal of this project was to meet many of the functional and non-functional requirements of an above-average reflow oven and make the product available to the UWB students and faculty. Our motivation was to have these higher-end features engineered into an affordable and reproducible converted toaster oven.

Summary of Project

Accomplishments:

In this project, we converted a consumer convection toaster oven so that it can be used for surface mount reflow soldering. We used a convection oven so that the heat is equally spread throughout the oven by the fan. To control the temperature, we used solid state relays switched through a printed circuit board controlled by a raspberry pi running a PID control algorithm. We read the temperature in the oven using two k-type thermocouples. The Raspberry Pi uses the temperature sensor to sense whether the oven needs to cool down or heat up according to a selected profile. To achieve the desired temperature, the pi will turn off and on the power to the oven's heating elements at a frequency no higher than 60 Hz.

We implemented a PID control algorithm to prescribe a percentage off and on to send to the SSR during each full second of operation, responsive to live temperature data inside the oven. Originally we hypothesized that finer control is not necessary or possible due to the expected thermodynamic characteristics.[1][2] We were able to confirm through testing that the level of control we have through this method is high enough precision to achieve our goals.

Additionally, we programmed a web UI deployed on a web server on the pi displayed on a local capacitive touch screen to allow the user to program custom temperature profiles into the oven, and select the appropriate profile according to the needs of any particular soldering paste.

We achieved all our functional goals for a build package cost of under $500, with a final implementation cost ~$460.

Limitations:

Our oven has two major limiting factors in comparison to high end expensive reflow ovens:

Our oven cannot follow temperature profiles which contain slopes in excess of 2.17 degrees C per second. Initially, we had a much lower maximum slope at 1.2 degree C per second, but made up the distance to 2.17 degrees with foil and ceramic insulation. The maximum slope a reasonable profile for soldering paste should ever contain is 3 degrees C per second, according to our research. [1]
We are nonetheless able to perform accurately for profiles for a wide range of soldering pastes given our current max slope, and our oven is programmed with several recommended default profiles.

Our oven's cooling is limited to between 0.5 and 1 degree C per second with the door closed. To mitigate our cooling issue, we programmed a small beeper and notification on UI telling the user when they may open the door to speed up the cooling process. When the user opens the door, the temperature in the oven drops precipitously, to the point that the control authority for the cooling curve would be determined by the thermodynamic characteristics of the particular board in the oven in accordance to how its component materials and surface area factors into its thermal conductance to the air. The maximum safe cooling curve for a PCB is expected to be 4 degrees C per second according to our research. [1]

Assuming the air around the oven is not significantly below freezing, we do not expect the thermal characteristics of a PCB to allow that rate of cooling to occur on the surface of the board, and our testing encountered no issues related to cooling rate.

User Requirements, Technical Requirements, and Deliverables

End-User Products

- **Converted Reflow Oven**
  The oven is repurposed to be used for reflow surface mount soldering. It is connected into a PCB that has a Raspberry Pi, temperature sensor, and solid-state relays. The Raspberry Pi acts as the brain and hosts the Web UI that will be used to adjust the temperature profile, which is displayed on a 5” capacitive touch screen.

- **User Documentation**
  - User Manual
    A user manual for the reflow oven and how to properly use it. This includes:
    - The different ways to adjust the temperature profile and how to use the User interface.
    - Safety/Instruction & Specs & Testing Documentation
      Include the reflow oven specifications such as maximum and minimum oven temperature and maximum and minimum temperature ramp. There will also be testing documentation on different tests performed and their results with instructions on how to operate the reflow oven safely.
    - Build Instructions including Components List
      This will include the components list and the exact amount used. It will include a step-by-step guide on how to recreate the reflow oven and the PCB layout and the Raspberry Pi code developed to control the oven.

Final Report

The final report includes how this project was developed and its evolution from the start. It includes the design decisions that we made and our reasoning behind them with any changes made from the proposal and a detailed product specification. The report includes records of the different characteristic testing performed and the results collected.

Description of Project

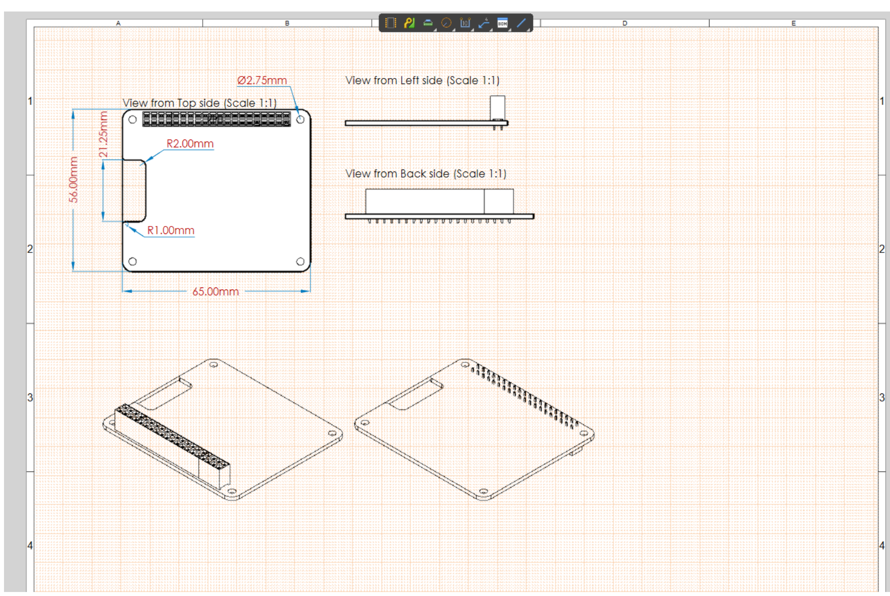
Approach to Engineering the Design Requirements

Research & Requirements Analysis

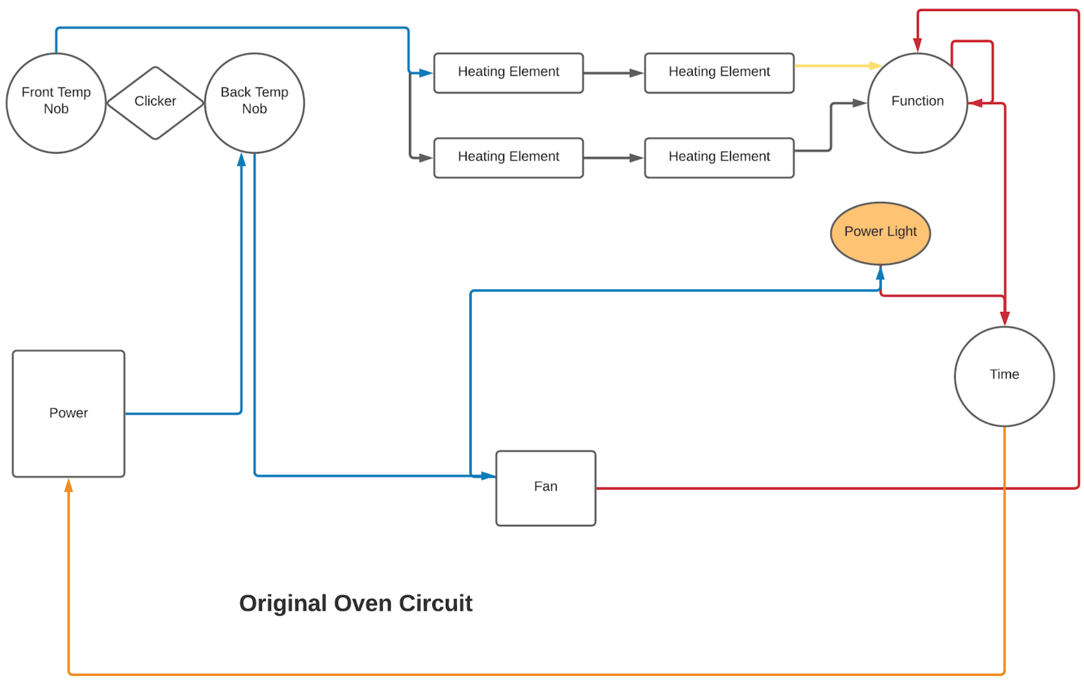
First we spent several weeks researching reflow ovens, with special attention to functional requirements. We also found several instances of similar cost efficient reflow oven projects undertaken by engineers for their own use from around the world. We watched videos on custom home built ovens, and took inspiration in particular in terms of the parts and budget that would be required. Several of our inspirations and most valuable research finds are linked in the Appendices section.

Design

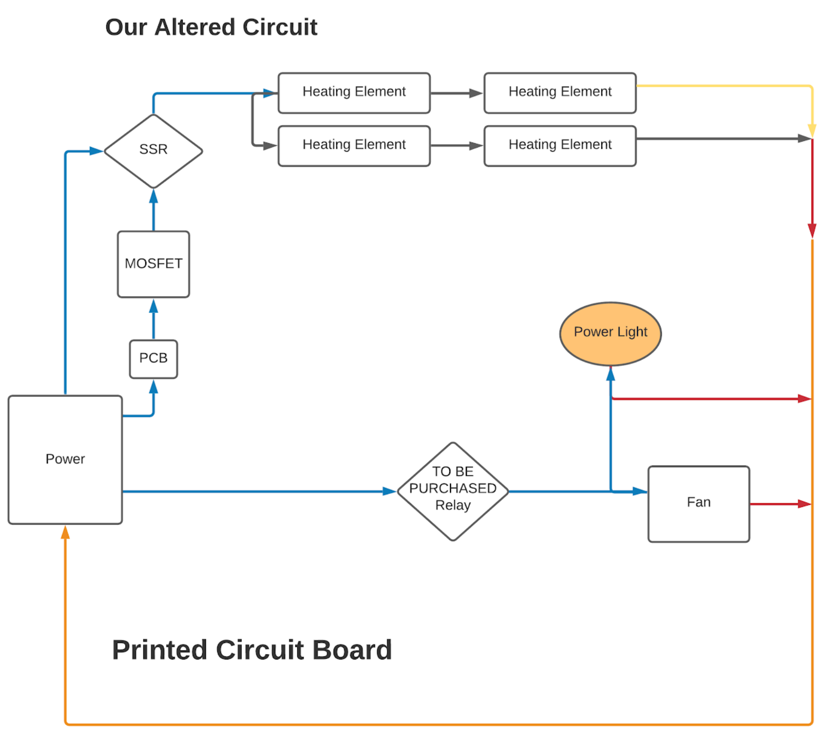
We started with circuit diagrams in LTSpice and sketches of our UI for initial prototyping. As we developed our understanding of the precise circuit required to integrate as parts arrived we were able to breadboard our circuit and test expected outputs and inputs with multimeters in the EE lab. With advice and initial training from our faculty advisor Dr. Rick Cordray, we were able to leverage Altium Designer to design the Printed Circuit Board serving the overall oven system. We also used javascript to program a webUI with an iterative design approach.



Initial stages of PCB design in Altium. This DWF view shows the planned mechanical dimensions.

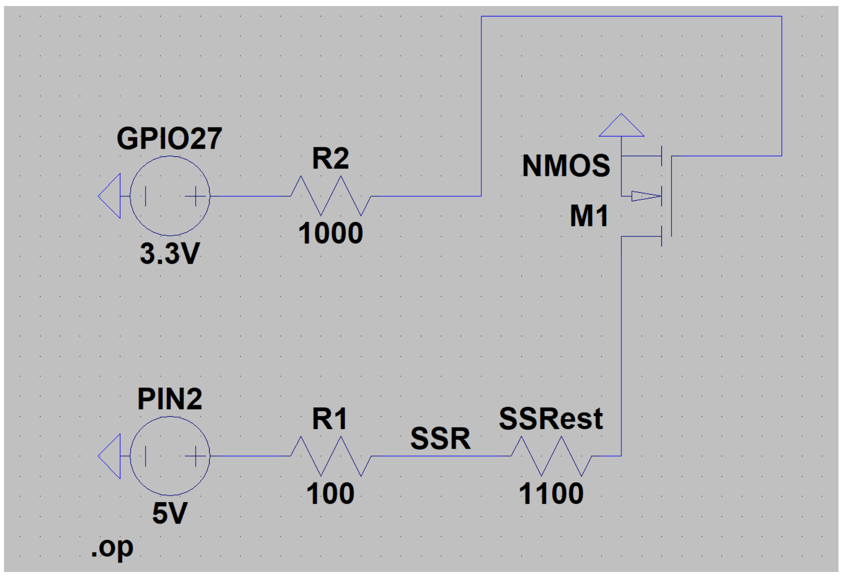


Our early stage analysis of the Original Oven Circuit upon disassembly

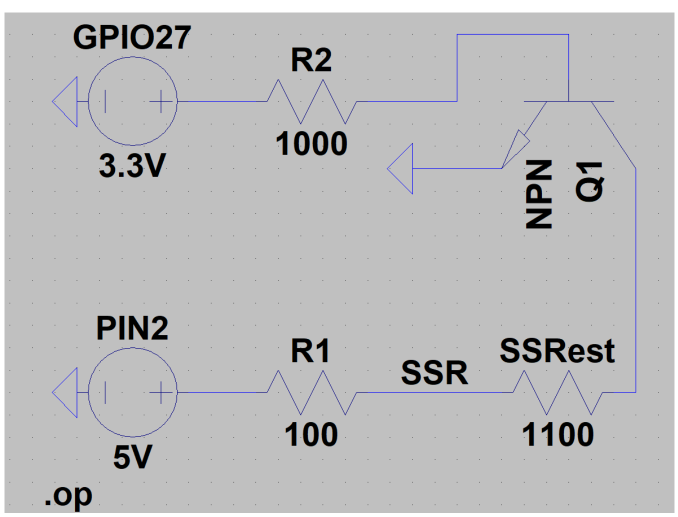


An early stage diagram of our altered circuit to control power to the heating elements

During our design, we finalized two redundant methods of controlling the SSRs to power the oven on the PCB, defining them as the RED or the BLUE build. Both methods of delivering current to the SSRs were built into the PCB as there were unknowns about undelivered parts. Once we had all our parts delivered, it was found the BLUE design was the overall more cost efficient and effective option. Providing the board with these two designed outputs allows for greater moddability in the future, should someone wish to use our PCB to switch power for two more 120 V devices using a raspberry pi. This could include an additional cooling fan, or an automatic door opening servomotor.



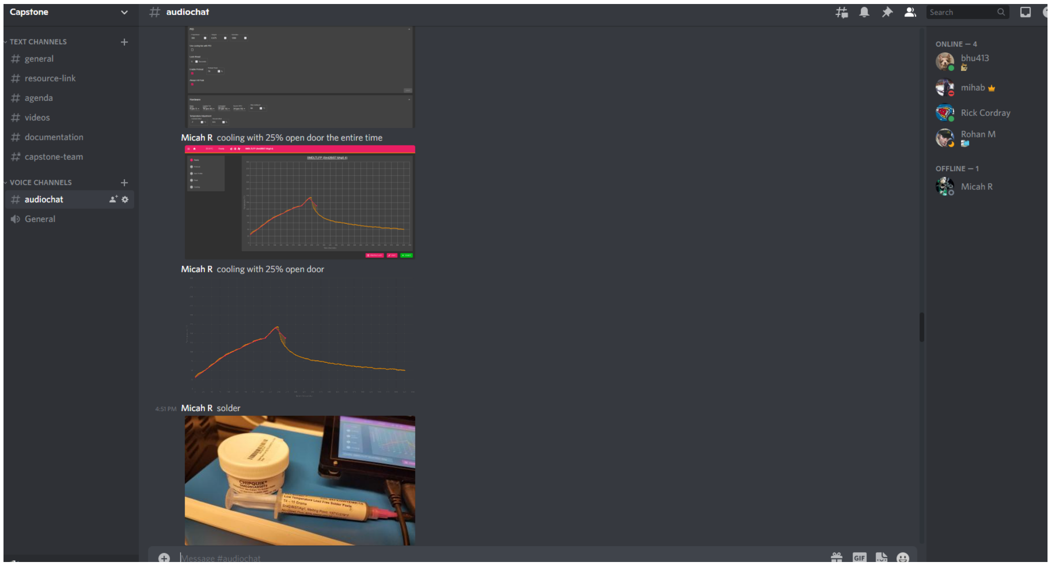
Preliminary Stage of LTspice design. In this stage, we were planning to use MOSFETs which ended up being replaced with NPN BJTs in our final RED design.



A later stage of LTSpice design, with a NPN BJT, final version of RED design.

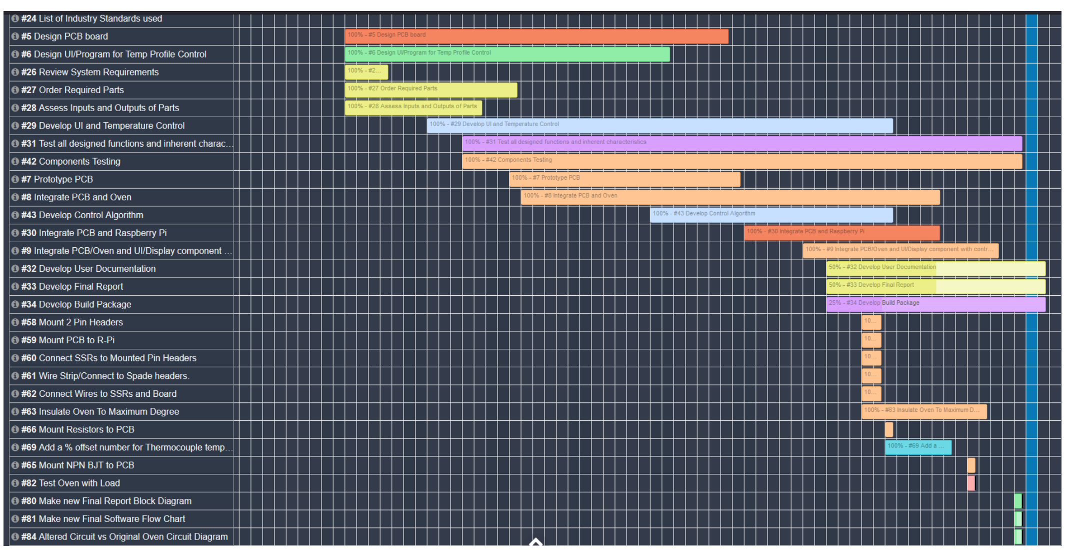
Planning & Task Management

An important phase of our project involved defining and planning out the major task to take place during our development cycle. We met several times a week using a dedicated Discord server to manage both our asynchronous and synchronous communication.

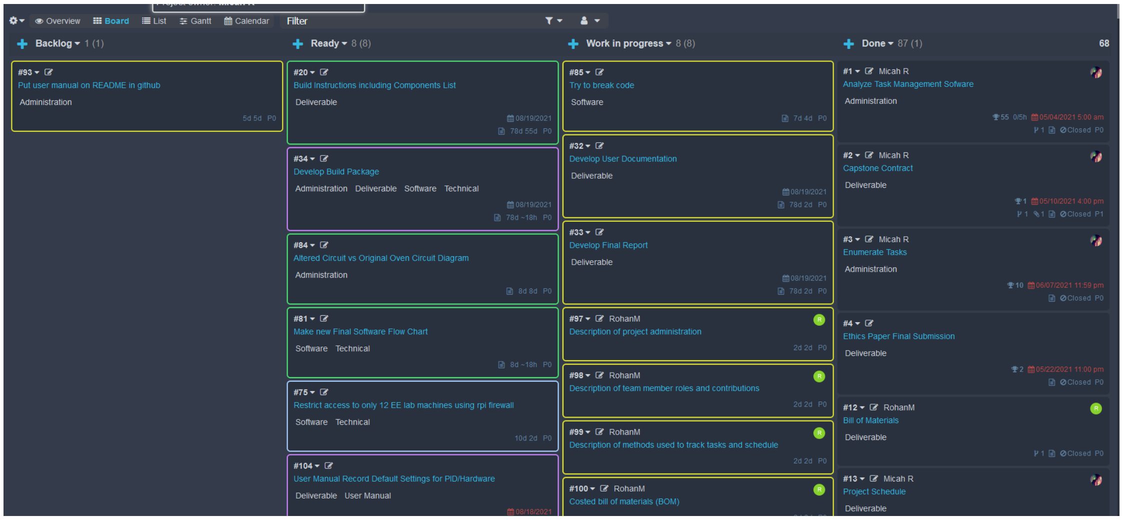


Discord Server showing late project stage communication

We decided to use the Kanban project engineering process as a basis for how we organized our work. We found a public open source implementation of Kanban project management, and hosted an instance on one of our UW student servers.



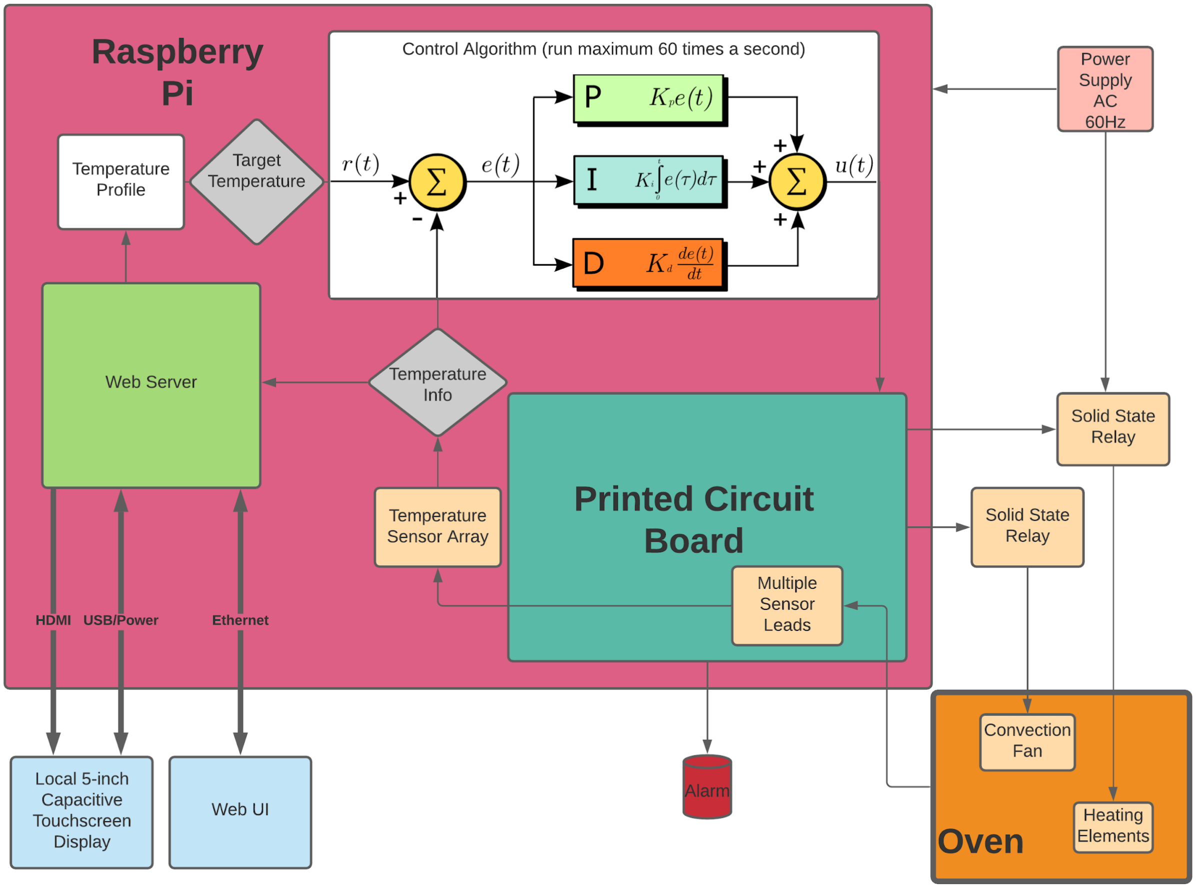
Gantt Chart View of some of our milestone Tasks



Kanboard Snapshot near end of Project

Using the Kanboard instance on our server we were able to define, assign, and link tasks, set due dates, and get visual representations of our workflow such as the highly useful for scheduling Gantt chart. As we moved through the various milestone tasks we defined child tasks for each challenge or piece of work that needed to be done, and were able to track and plan for dependent tasks, anticipating impacts on the long term schedule.

System Block Diagram with Item Descriptions



The block diagram above shows what components are connected to each other as well as the boundaries of each component. Connections to and from the external components are indicated with arrows that go to the designated internal components that they interact with. The PCB is shown to be encapsulating the Raspberry Pi since the Raspberry Pi will be mounted onto it for an all-in-one design. One thing to note is that the control algorithm is not actually a physical entity. Instead, it is there to show the inputs and outputs of the Raspberry pi that enable it to control the temperature of the oven.

Printed Circuit Board

The printed circuit board provides an elegant way to connect all of the components to the Raspberry Pi instead of using jumper cables and a breadboard.

Raspberry Pi

The Raspberry Pi is essentially the brains of the oven and is in charge of running the UI, polling the temperature, and calculating the amount of time for the heating element to turn on so that it can follow the temperature profile that the user has selected.

Web Server

Because our UI is web-based, the web server is in charge of providing the UI and updating it with the current temperature. Depending on pending processes and conversations with UWIT, the web server can also allow users to connect remotely by accessing the web server’s IP address and port.

Touchscreen and Web UI

A 5-inch, capacitive touchscreen display is installed with the oven so that users are able to control and monitor the oven locally using the web UI that is displayed on a kiosk mode browser. The touchscreen allows for easy custom profile creation with draggable points on a graph.

Control Algorithm

The PID algorithm on the web server backend is used to calculate the amount of time that the solid-state relay will turn on for each next second of operation, given the current target temperature, and the current actual temperature. Users may define new PID constants if they wish in the UI.

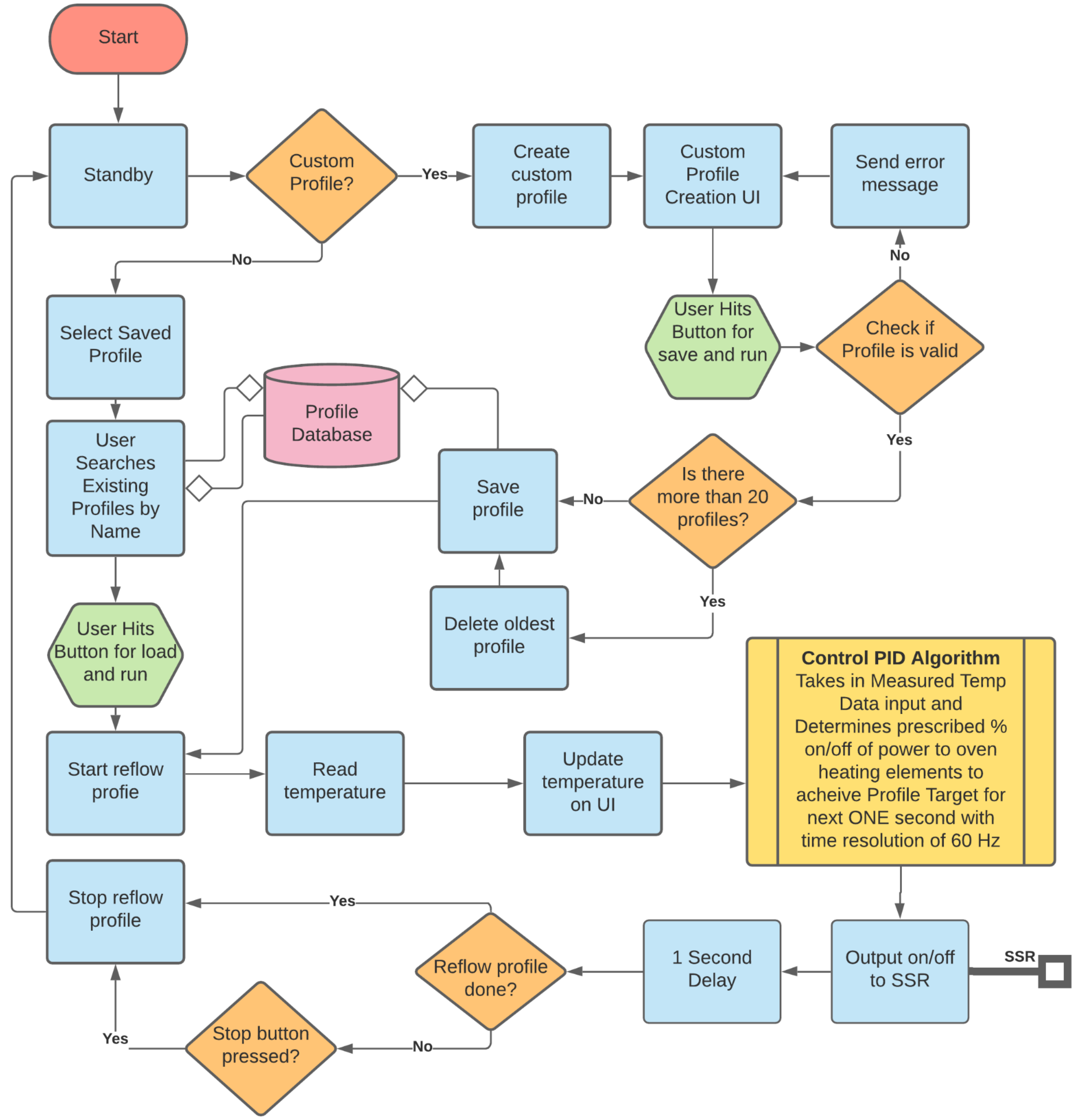
Solid State Relay

This component allows us to control the heating elements using a low voltage. Because it is solid-state, we are able to switch it on and off at a high speed for more precise temperature control.

Temperature Sensor Array and Leads

There are two temperature sensors spread in parts of the oven to allow for accurate reading of the temperature.

Software Flowchart



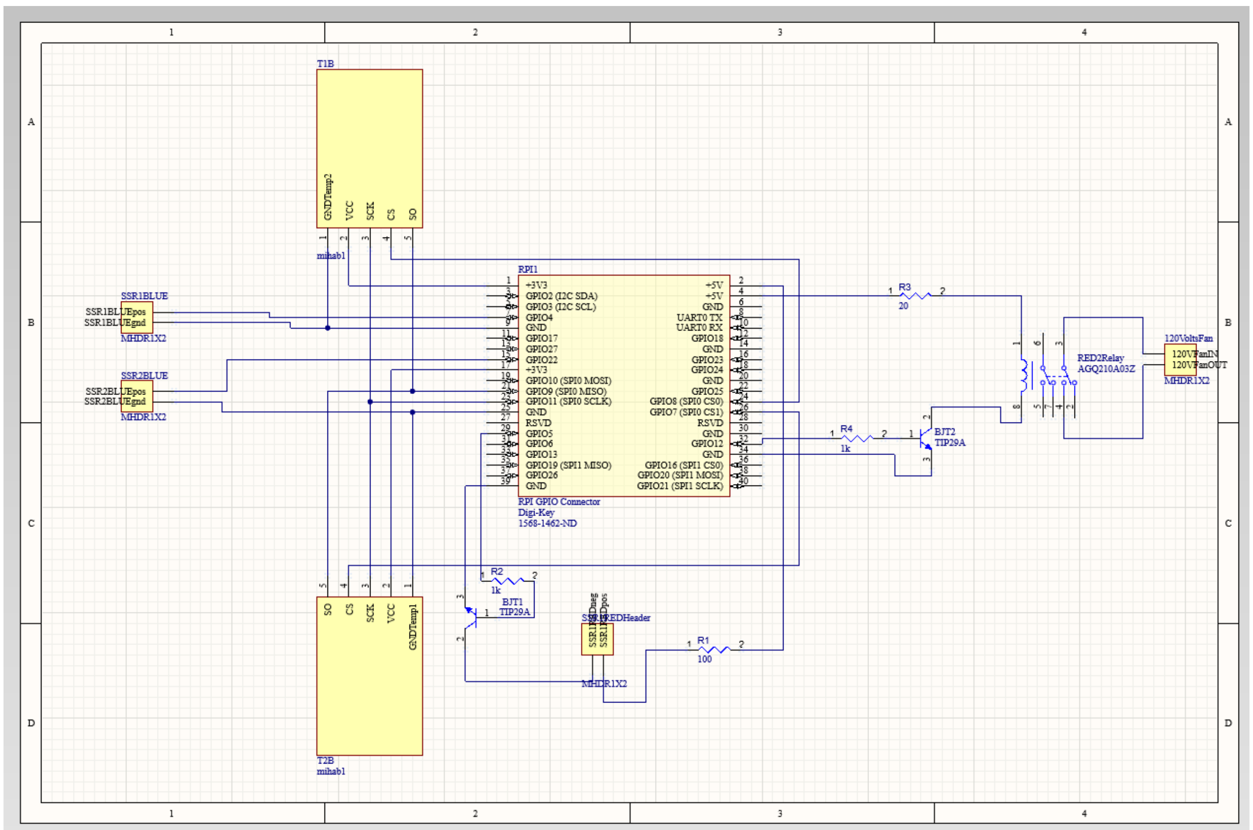
This flowchart describes the actions taken by the web server and UI as the user interacts with the oven. The user can choose either to load a pre-existing profile or make and save a custom one using the profile creation page in the UI. After successfully loading and starting a profile the program will continuously check and maintain the temperature according to the temperature profile duration using the control PID algorithm. The user is able to stop the program at any time while it is running the profile.

Integrated System and Packaging Details

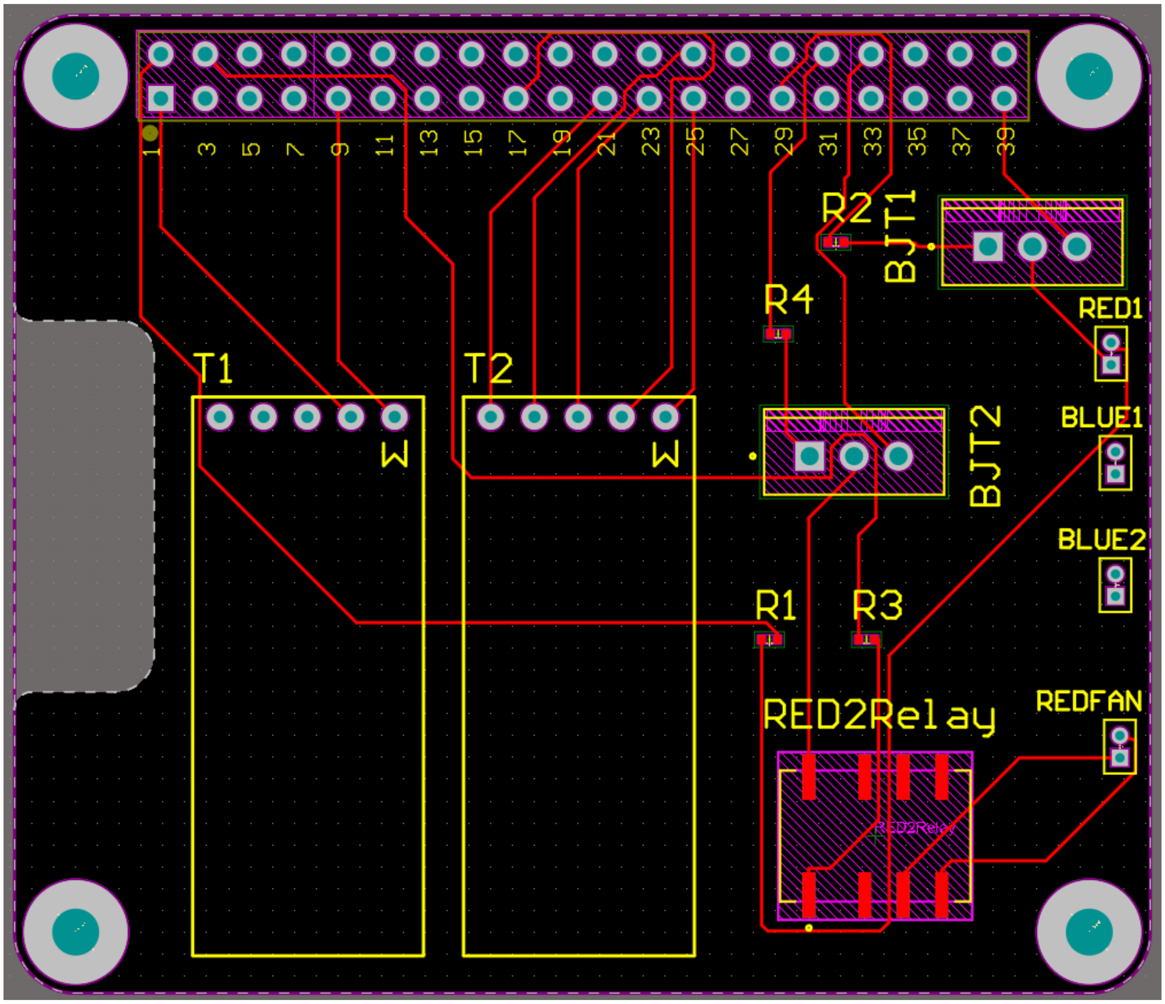
Final PCB Design

Design Software Used

We were trained in the use of Altium Designer, a powerful circuit board design tool, by our Faculty advisor Dr. Rick Cordray. We obtained student keys from the publisher and began translating our LTSpice designs into Altium's more powerful design layered system. We were able to select precise manufacturer parts with predefined footprints and put them on our board.



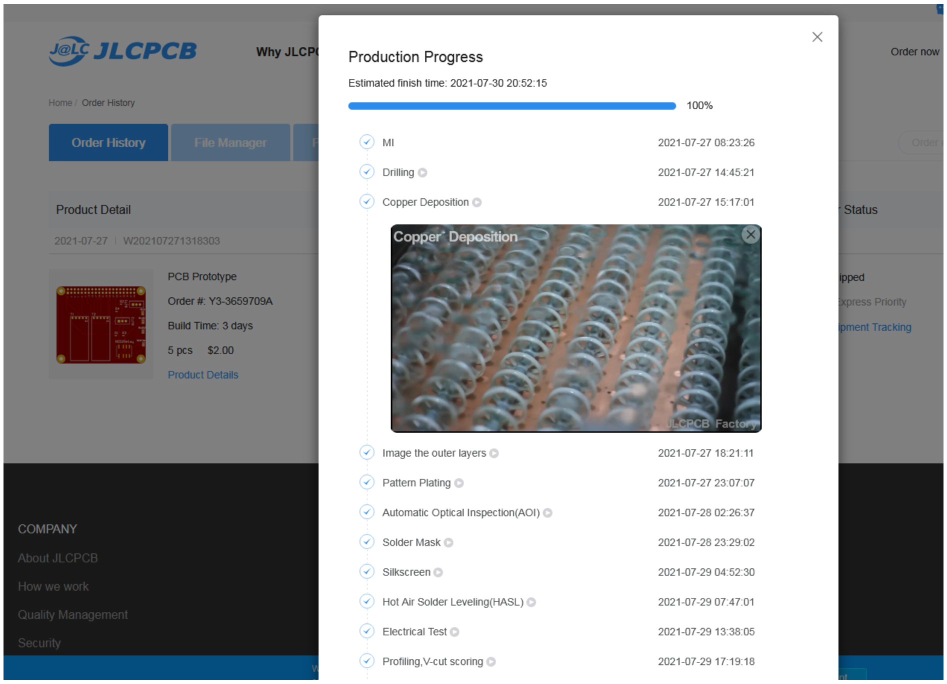
Schematic view of our final PCB in Altium Designer



Final PCB view in Altium Designer

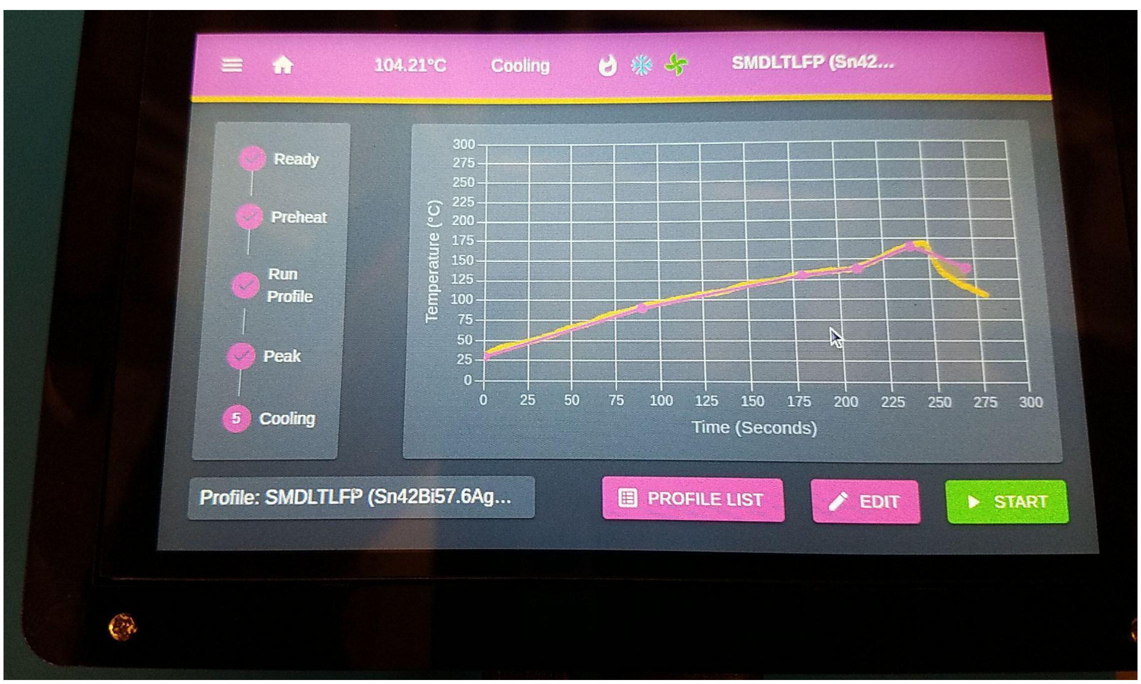
Manufacturing process

We worked with a company in Hong Kong known as JLCPCB to manufacture the PCB. During the process of collecting the gerber files and sending them for manufacture an error was discovered wherein the drill holes for our PCB were slightly misaligned with our contact pads. The engineers at JLCPCB pointed out there was a misalignment, and we traced the error to two different versions of our files saved at different stages of development. We regenerated the drill hole file to match our correct design, and were able to start the PCB manufacturing process on schedule.



An image of the manufacturing process tracker from the JLCPCB.com website.

Final Software Deployment information



5" Capacitive Touch screen display attached to oven showing Web UI

The software for our UI application is available on github at this location:
https://github.com/bhu413/reflow

The web server for the UI runs on the same Raspberry Pi that is controlling the temperature in the oven. Depending on pending conversations with UW IT, the web UI may be accessible from student's lab computers in the same room as the deployed reflow oven. However, all control of the oven is designed to be operated perfectly exclusively from the local touchscreen. The web access option is simply an additional bonus feature which may be applicable if students are running long profiles they may want to keep an eye on from their lab bench rather than standing around the oven.

Assembly and Package information

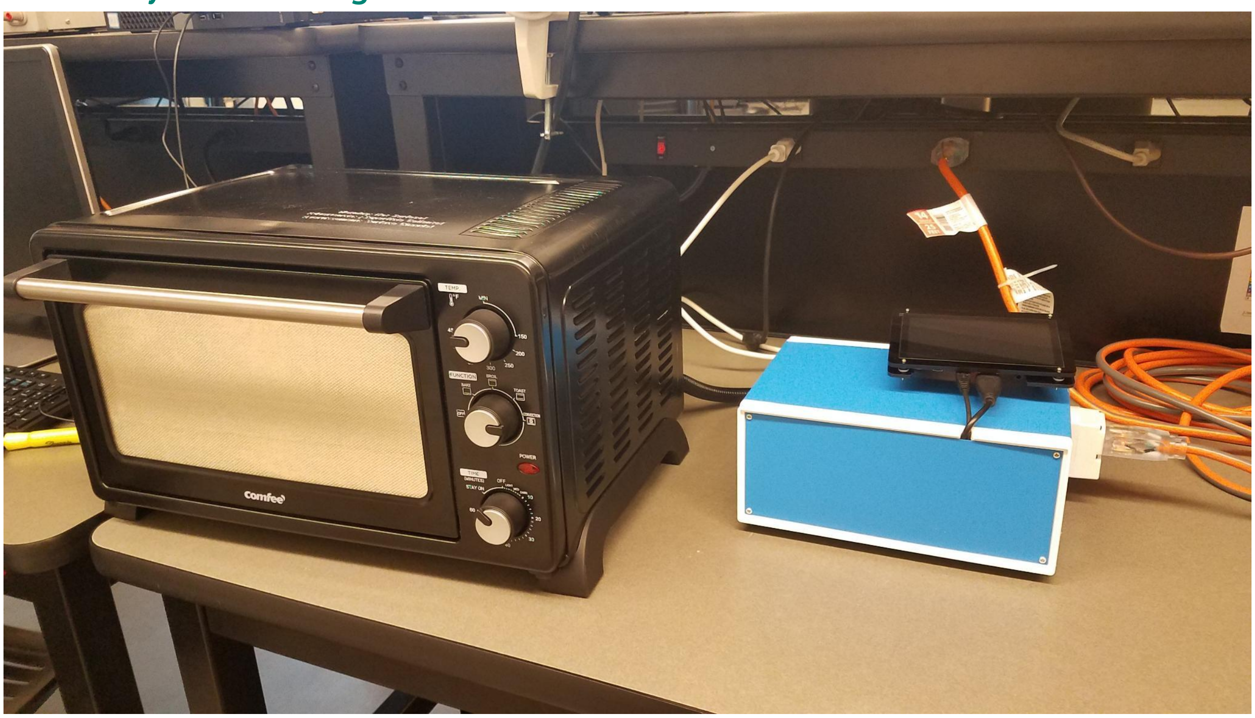
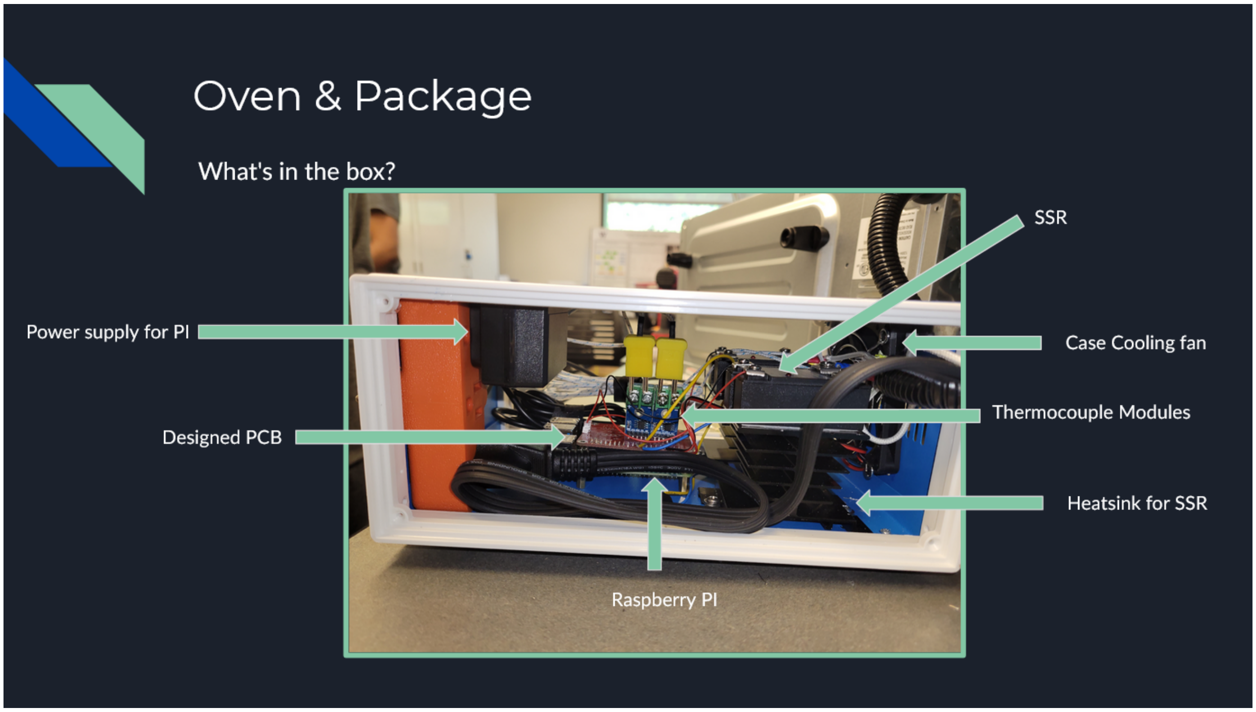


Image shows the reflow oven, with blue controller box and touch screen display on the right.

To assemble our reflow oven controller case, we drilled holes in the metal walls for each point that needed to be secured inside the case. For example, the SSRs required holes drilled into the casing along with nuts and bolts to secure the heat sinks to the floor of the case.



A slide from our final presentation showing the components inside the control case.

We protected the wires coming from the oven to the case using a plastic wire jacket sleeve, and drilled a hole in the case for the sleeve to come through with all the sensor and power wires.

Power supply to the oven is run through a 15 A rated 3 hole extension socket, a 15 A rated switching outlet on the exterior of the case plugged into a 14 gauge single outlet extension cord. We calculated the load to be ~13 A.

Requirements Validation

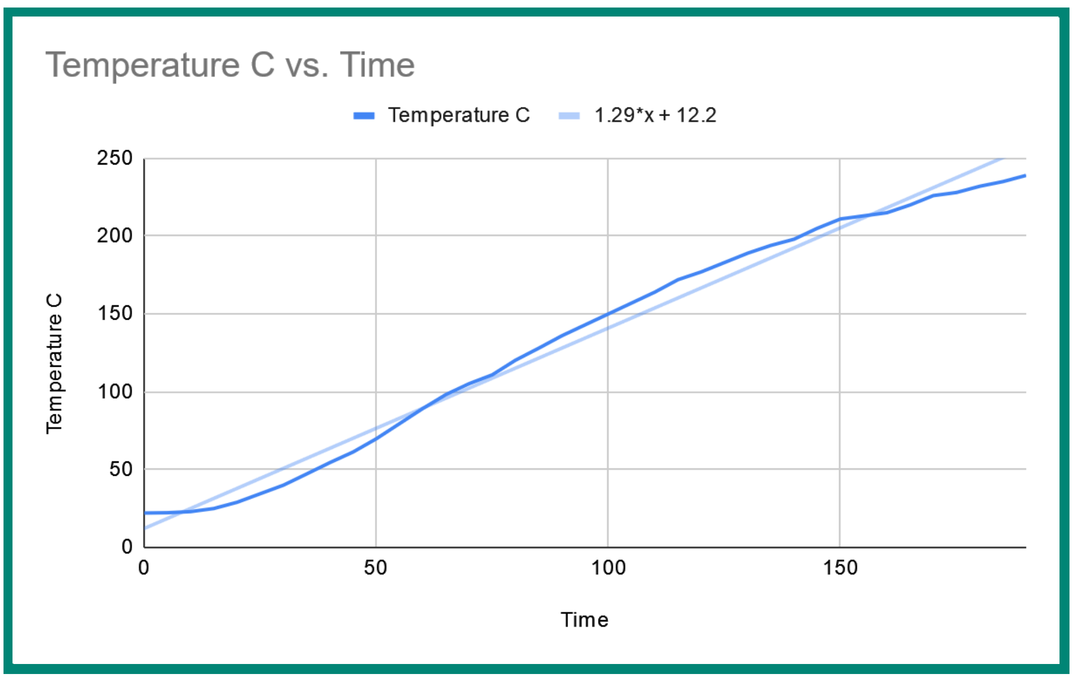
Testing Functions

Procedures

Prior to final load tests, where the function of our oven was validated by simply using it, we tested the capacity of our oven to heat to a specific slope to get an idea of what range of slopes we could support.

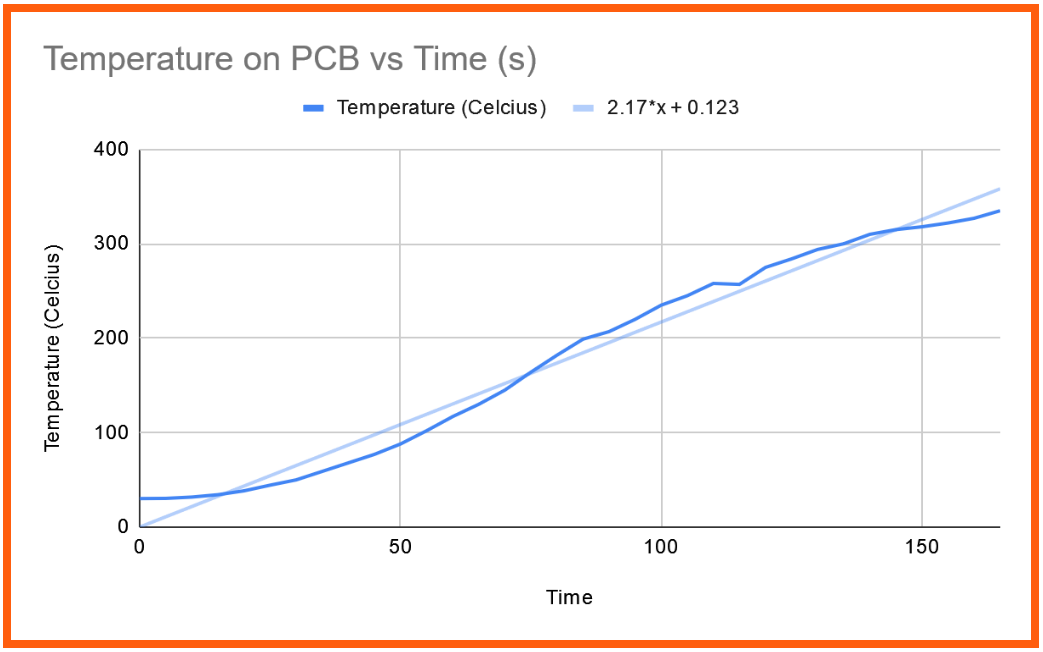
We collected data using a separate thermocouple at the location of the PCB so as to simultaneously test curve slope and correlation between PCB temperature and temperature at the location of our two thermocouples hanging in the air above the PCB rack.

While the temperature data history for a profile is stored during the run in the UI, the more important PCB local data was just being reported to a thermometer connected to a thermocouple taped to a PCB location. So we set a 5 second interval timer and called out the temperatures read, inputting them into a spreadsheet to produce the graphs below.



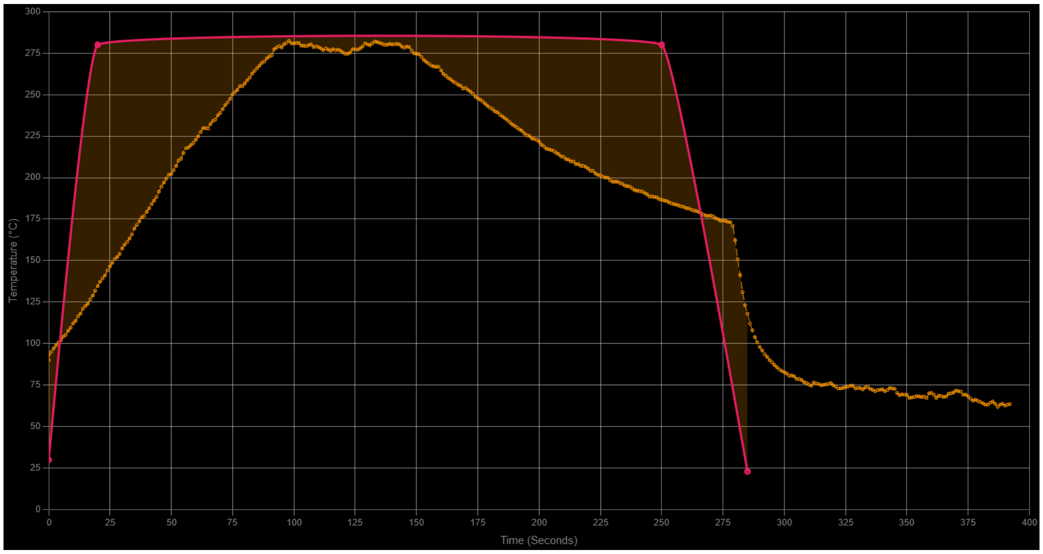
Our initial maximum slope with the heating elements being sent 100% uptime by the PID control algorithm was limited to between 1.2 and 1.3 degrees C per second.

After testing determined the need for insulation, we installed foil and ceramic insulation in the ovens. Through this we were able to achieve a higher final maximum slope.



Maximum Final slope of 2.17 degrees C/s on average. Notably, the slope is higher in the 100-200 degree C range.

Recorded data on the control UI was accomplished by setting profiles to unrealistically high slopes and watching the actual temperature curve. In the below image, the max temp is not actually missed at the actual temp curve's peak. This was an earlier version of our UI which curved the profile from point to point, in this case unintentionally implying the target temp was higher in the middle than it actually was.



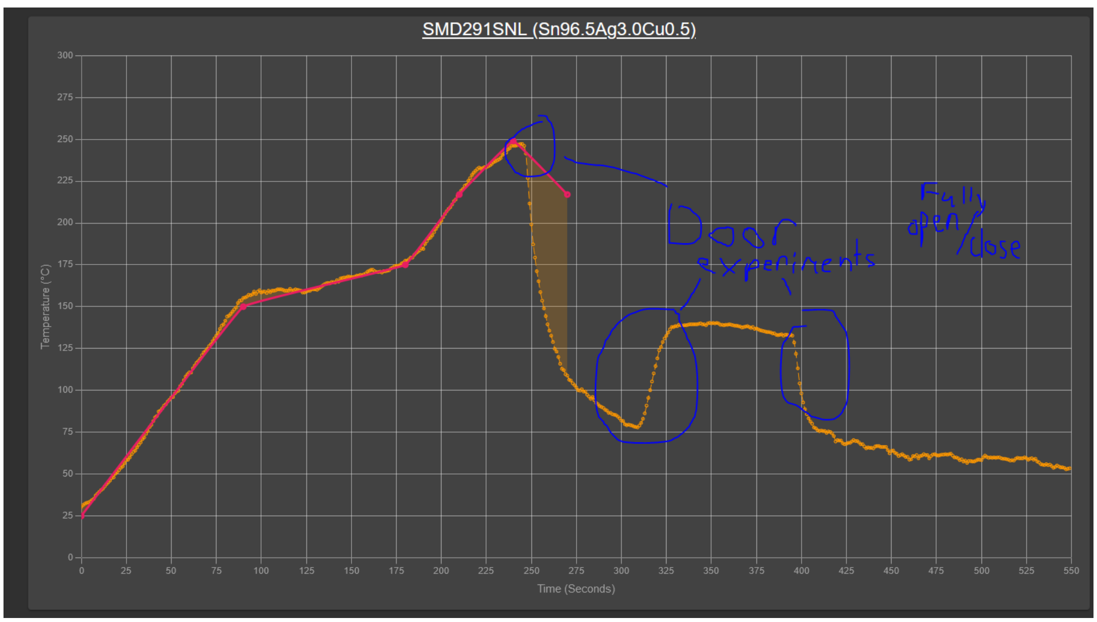
Above, the Red line represents the temperature curve, and the yellow line represents actual temperature followed over time in a test.

Results

Once we hit our target temperature slopes of at least 2 degrees per second by adding extra insulation, we began testing our cooling curves. We found we can reliably fall by 1 degree per second between 300 and 150 degrees C and 0.5 degrees per second between 150 and 25 degrees per second.

Soldering paste compound profiles often call for a PCB cooling of around 2-4 degrees C/s. We are able to follow a wide range of profiles with exactness on the ramp up using our PID algorithm from Raspberry Pi outputting through our PCB to our SSRs controlling power to our Oven, during flux activation soak, and the reflow.

However, the cooling was too slow given the extra insulation. To combat this, we added functionality to the user interface which prompts the user to open the door. We tested the various effects this had on the temperature curve as shown below.



In this test, we opened the door 100% at peak temperature, and closed it again completely after 50 seconds.

We knew that while our temperature readings in the UI showed a massive drop off, the actual cooling on the PCB would be subject to the thermal properties of the PCB itself dissipating heat to air depending on the materials in the component parts and the surface area of the board. We postulate that with our open door method we expose a typical board to cooling within the 2-4 degree range, and our testing revealed no negative side effects to this cooling method.

Evaluation of Performance

Comparison to deliverable goals

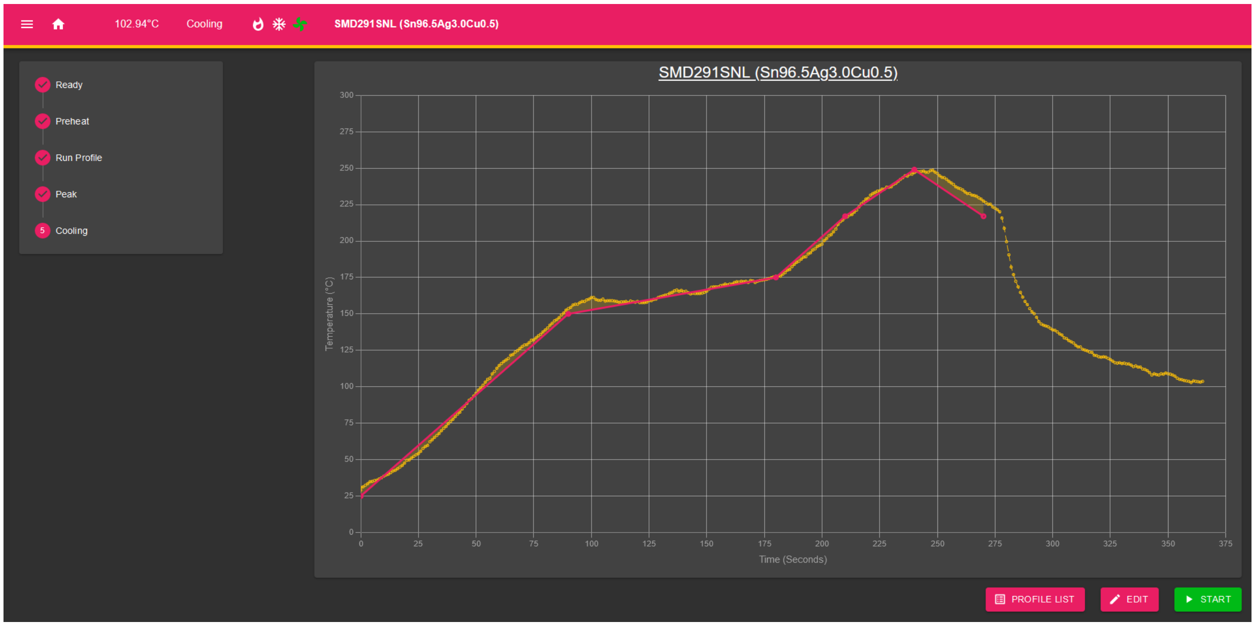
One final set of tests we did not have the time nor covid protocols in place to run which might have led to more development are Usability tests. We would have liked to invite uninitiated folks to attempt to use our oven while taking notes on their decisions and presumptions about the information being presented to them. This might have led to changes to our UI to make sure there aren't misleading or confusing elements, as one of our design goals was a quick and easy to use interface. In the absence of that test, we nonetheless believe we have achieved that goal as best we can perceive it.

As our Oven's deliverable goals relate to the Temperature control, we are able to serve a great deal of reflow soldering paste compounds given our coverage of the possible ramp spectrum. As mentioned elsewhere in this document, the expected range for profile temperature ramps is 1-3 degrees C per second. As we are able to achieve anything between 0 and 2.17 degrees C per second, we are able to cover many different types of pastes, and all of those we found available in the UW EE lab wherein we expect our reflow oven to be installed.

As for cooling, we believe that while prompting the user to open the door is a viable solution, additional development is possible here for our project. Some suggestions for any future project group working with our design include installing a servo motor to automatically open the door by 10%. This would be in place of our current solution, which is to simply request the user do so themselves. This would also afford the opportunity for more precise cooling control as well as more automatic control, as the door could open and close slightly depending on the dictates of our PID algorithm. Another suggestion is a stronger convection fan(which is already controlled by our pi), which may allow an increase in power during door opening to greatly affect the air cooling of the pi.

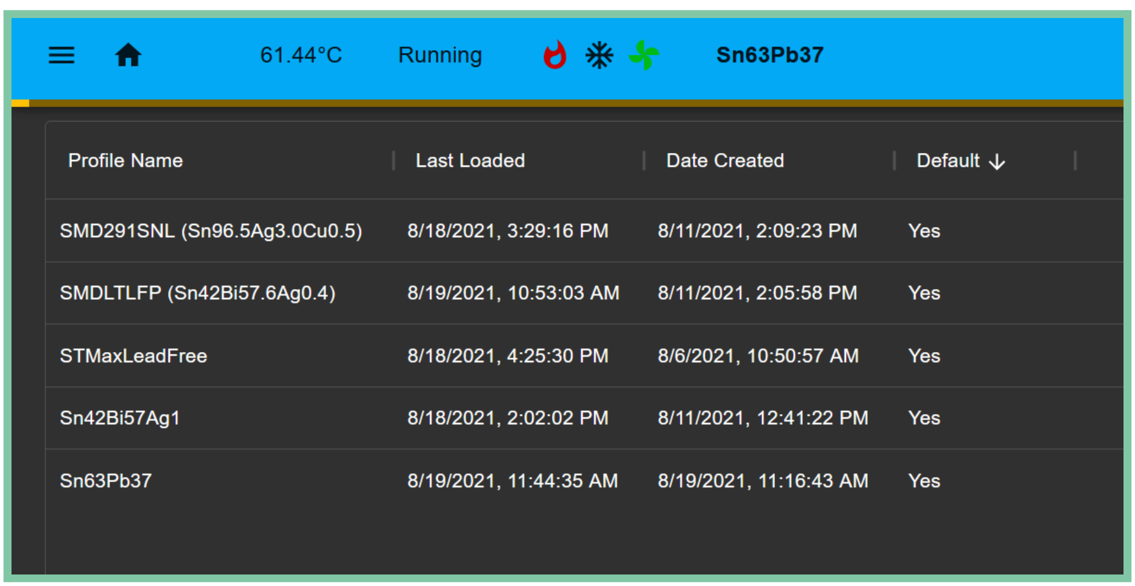
These additional controls are already possible given our project PCB, using the RED design outputs to control functions of another fan and motor, so no additional changes would need to be made to our PCB design.

While greater development is possible, we also believe we have met the practical performance requirements of the cooling side of a reflow process as well, as we do not recommend the oven be left completely unattended, and opening the door does adequately meet the cooling needs of a PCB post reflow.



One of our final loaded Profile tests, showing the accuracy following the profile.

As for the total functionality of the system, we tested it with our own PCB, mounting the parts necessary for our RED build successfully in our own oven.



Profile Name	Last Loaded	Date Created	Default ↓
SMD291SNL (Sn96.5Ag3.0Cu0.5)	8/18/2021, 3:29:16 PM	8/11/2021, 2:09:23 PM	Yes
SMDLTLFP (Sn42Bi57.6Ag0.4)	8/19/2021, 10:53:03 AM	8/11/2021, 2:05:58 PM	Yes
STMaxLeadFree	8/18/2021, 4:25:30 PM	8/6/2021, 10:50:57 AM	Yes
Sn42Bi57Ag1	8/18/2021, 2:02:02 PM	8/11/2021, 12:41:22 PM	Yes
Sn63Pb37	8/19/2021, 11:44:35 AM	8/19/2021, 11:16:43 AM	Yes

Shown above are the default profiles preinstalled on the oven.

In addition to the default profiles, the user may program any profile which does not have a temperature ramp exceeding 2.17 degrees C per second and expect the oven to follow it with exactness.

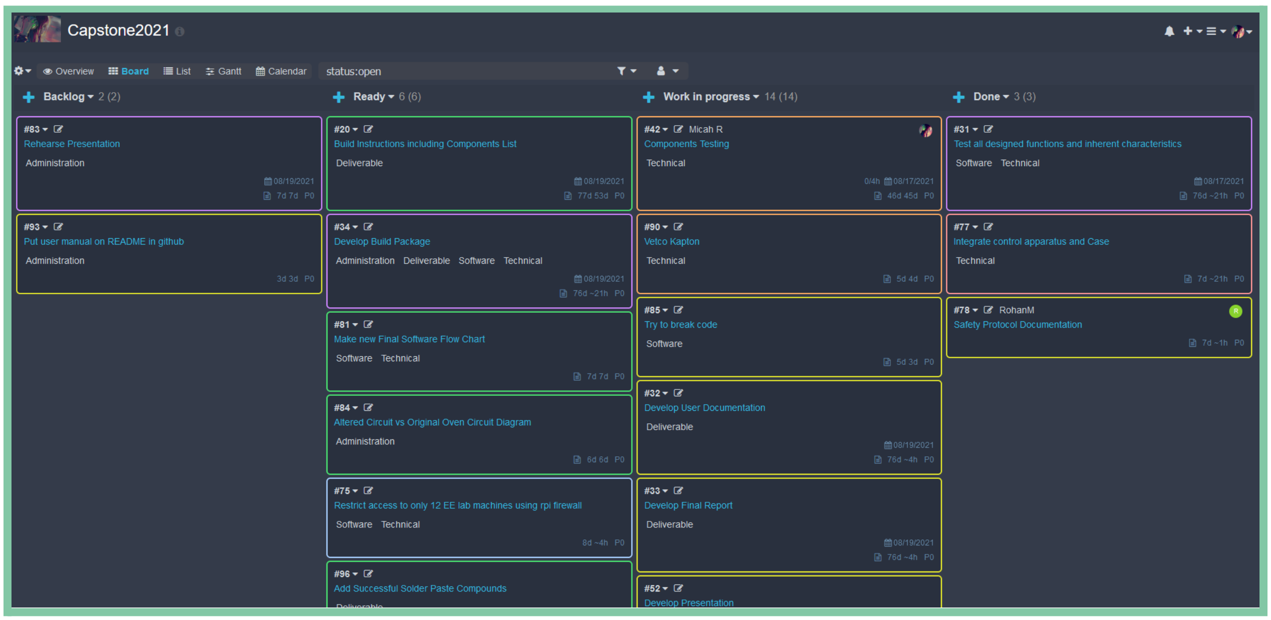
These features of temperature precision, information display, and user programmability stand up well to our target features found in a $3800 oven the University labs recently acquired. We have achieved our project goals with a final build cost of $460, and a total project development and build cost together of $533. Since many of the parts like the PCB we purchased came in bulk, we believe our build could be replicated for under $400, and improved upon for a similar final cost as our actual spent budget.

Project Administration

Schedule/Task Management

Kanban Board Tasks

We decided to use a kanban engineering process to manage our tasks during this project. We found an open source implementation of Kanban called Kanboard and hosted an instance on Micah Rice's UW student server. All project members were able to login and see the Kanban board to create new tasks, move them from column to column, or assign them to a particular member, among many other powerful features. Kanboard can be found at kanboard.org, as well as at https://github.com/kanboard/kanboard/releases.

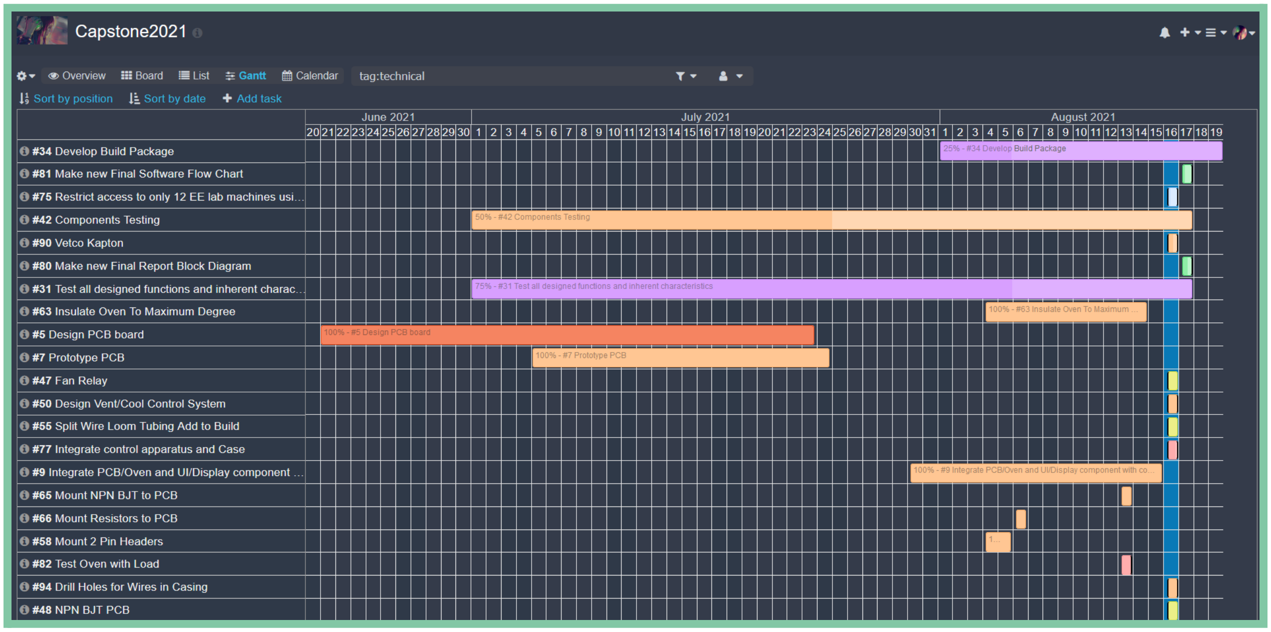


Above is an example view of the Kanboard late in the development cycle.

Kanboard Gantt Chart

It was important for us to plan ahead to our major project milestones and also to understand how each task impacted the next, and identify any dependencies. At the advice of our Faculty Advisor Dr. Rick Cordray, we decided to install an add on to the Kanboard software hosted on our server which added a Gantt chart functionality.

We used this to plan ahead, and understand how a delay in any one task was going to affect future tasks.

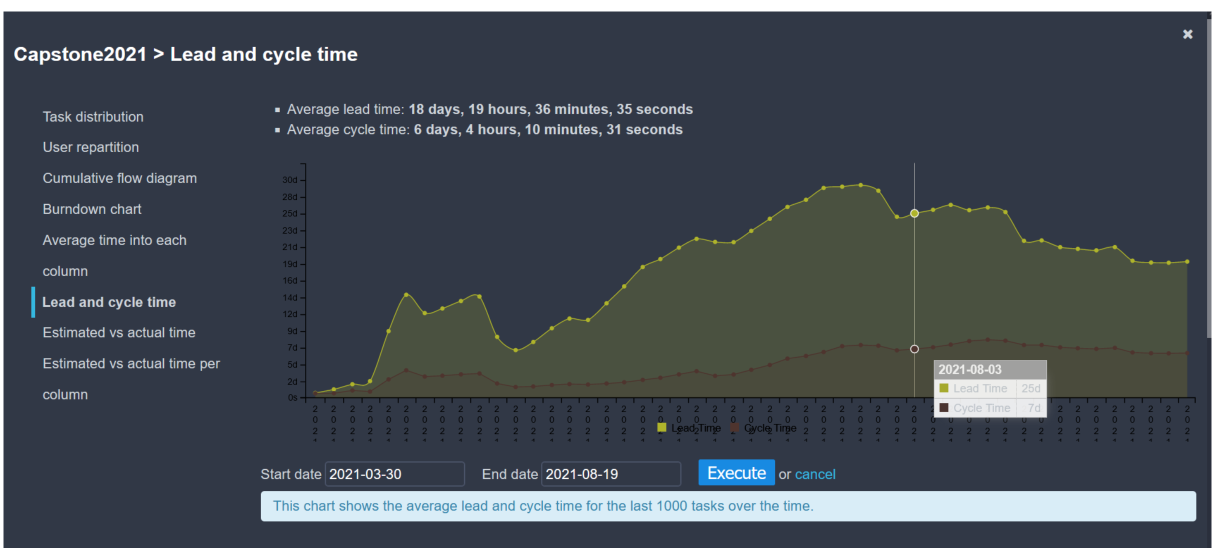


Gantt view is filtered to show only tasks that were tagged "technical".

We categorized tasks according to their purpose. In addition to technical, other tags included Deliverable, Software, REDBUILD, and Documentation.

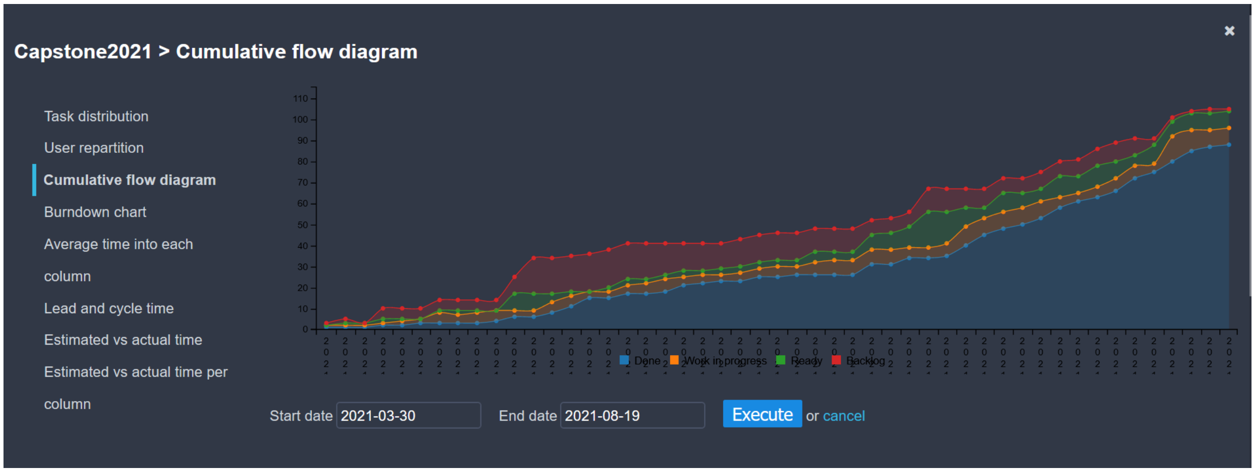
Kanboard Analytics

We can now look back and reflect upon the various characteristics of our project management through the lens of Kanboard Analytics.



Lead and Cycle Time for tasks entered in the duration of the entire project

We can see from the Lead and Cycle Time chart that we typically knew of a task that was necessary to complete for about a month in advance of its completion. Once a task was begun, it typically took a week to complete.



Cumulative flow diagram for tasks undertaken in the duration of the entire project

The cumulative flow diagram shows the distribution of the tasks in the Backlog, Ready, WIP, and Done columns throughout the project. As the project nears its end, the Ready and WIP columns begin to eat up the remaining backlog, and the burn rate of tasks increases.

Team Member Roles and Contributions

Brandon Hu CSS

For this capstone project, my role was creating the backend and frontend for the oven. I was in charge of prototyping the user interface design as well as gathering the functional and non-functional requirements that need to be implemented in the software. Regarding the development of the software, I used Node.js and React because of the vast amount of libraries it provides. I was able to use them together to create a whole application that contains the user interface as well as the oven control.

Because we were dealing with high heat, I had to make sure that the controller software was robust and accurate. I implemented features such as preheat-to-first-node, always-hit-peak, and fast forward to ambient temperature point. These features were used to mitigate edge cases and further improve the accuracy of our PID algorithm.

As we got further in the quarter, I implemented feature requests from the Kanboard as they came in. I also communicated with UW IT to make sure that our remote access feature was approved and informed the team what needed to be done. During the fabrication process, I helped out the team with installing insulation as well as characterization testing.

Micah Rice CE

As a Computer Engineer, I have training in both Computer Science and Electrical Engineering concepts. Thus I was in a multi-role flexible state, able to act as a sort of liaison between the EE work and CSS work, ensuring a common understanding of how the software produced by Brandon and hardware portions produced by Rohan, Mihertab and myself interact. I installed and hosted the Kanboard, which I leveraged to define, track, and assign tasks, building and managing the project schedule, and communicating team short and long term project goals. I wrote agendas for our weekly meetings in accordance with the tasks and milestones defined on the kanboard, and composed purchase orders for our final build phase.

I participated in all PCB design tasks along with the rest of the team, creating and sharing design files for the board and initial circuit diagrams along with the team.
I performed a design review for our PCB and corrected issues before sending it to Manufacturing. I performed build tasks during assembly along with the team, and designed and performed testing tasks with multimeters, breadboard, and final components during project design and validation stages.
I wrote a great deal of documentation & technical writing and was responsible for several large sections of this final report, as well as parts of the User Manual.
I asked a lot of questions, investigated a lot of answers, and learned about practical application of classroom knowledge.

Rohan Mathew EE

For my role as an Electrical Engineer, I was working alongside Mihretab on the PCB design as well as making sure that everything would function properly together and mathematically makes sense based on the parts we decided to use. I was focused on making sure that all components will be able to work together in the circuit and that nothing would be damaged in any way. Conceptually making sure to research everything we used so we all know how to properly incorporate it in our project and have it perform the desired task. I also helped breadboarding circuits for tests.

In addition, I was making sure to calculate all the necessary values for certain components for our circuit. I was also a part of both taking apart the oven and putting in the new parts as well as adding new pieces to the oven. Making sure that our designed PCB circuit was wired and set up identically to the schematic we made on Altium. Also, when testing I made sure to record our data and to make note of anything that stands out or if something is not right. Recording data on excel and taking pictures so that we have something to reference to whenever we refer to our results.

Additionally, when building the box and adding in our parts for the oven, I helped out in adjusting and placing a lot of the parts. Making sure that the right components were drilled or super glued on properly. I made sure to have everyone aware of the dangers when using new tools and running the oven at dangerous temperatures.

Mihretab Desta EE

In this project, as an Electrical Engineer, my focus was more on the electrical and tasks related to circuit design.

Once we decided to use Altium designer, I was able to set up Altium 360 so we can all contribute to the designing aspect at our own pace. One of my main contributions was designing the printed circuit board on Altium Designer.

I was able to make a footprint for components when they weren't able and to incorporate them into our PCB design. I made the PCB design functional and made sure all the connections were correct and we don't have any short or traces going to the wrong pin.

Aside from that, I checked all the traces were as designed on Altium once the PCB was delivered. I participated in calculations of current drawn by our circuit and made sure our system was not drawing a greater amount of power than our system can handle. I also contributed a great deal in the construction of the oven.I participated in the assembly of the PCB and the components we wanted soldered on it.

I contributed to building the oven and installing the insulation that we needed. I was able to voice my concerns and also share my ideas to improve our design and challenge our understanding. I was able to use the skill I learned in class and apply it in the project design.

Bill of Materials

Id	Description	Quan	Supplier	Supplier Part #	Mfr. Part #	Price Each	Totals
1	Rasberry Pi 4 GB with heatsinks	1	Amazon	B07TXKY4Z9	CanaKit	$74.99	$74.99
3	Convection Oven: Preference 1	1	Amazon	B0847RBPNW	COMFEE - CFO-CC2501	$59.99	$59.99
4	Temp Sensor	3	Amazon	B01HT871SO	HiLetgo	$8.49	$25.47
5	5 inch capacitive touch display	1	Amazon	B082F3K84X	Jun-Electron	$55.99	$55.99
6	MicroSD 32 GB	1	Amazon	B07B98GXQT	MB-MJ32GA/AM	$7.99	$7.99
8	SSR Solution for Fan and Heating Elements	2	Amazon	B08GPJ1V2J	CGELE SSR-25DA	$9.90	$19.80
9	Heat Sink For Relay	2	Amazon	B091HQL9TM	CGELE 1PCS Alum Heat Sink SSR	$6.59	$13.18
10	K - Type Thermocouples, high temp capacity	1	Amazon	B00OLNZ6XI	SZZJ INC 1	$15.99	$15.99
11	Spade Connector Kit	1	Amazon	B08BZ9J757	TICONN 25170000	10.95	$10.95
12	Female Pin Headersn for R-Pi to PCB	1	Amazon	B07DNHS2SJ	Wallfront Ean 0763741503994	9.59	$9.59
13	PCB Print/Deliver	1	https://jlcpcb.com/			15	$15.00
14	High Temp Insulation 16 Gauge Wire	1	Amazon	B07W3D3SL9	Bryne AWG16 10Ft	13.88	$13.88
16	Foil for Insulation	2	Amazon	B00029KC2K	Thermo-Tec 200-10045	25.33	$50.66
21	Extension cord 3 socket	1	Amazon	B000Y4DXLA	Coleman Cable 2451	20.82	$20.82
22	case	1	Amazon	B08BS39P69	Zulkiit Box 9.8 x 7.5 x 4.3 Inch	18.99	18.99
23	Wire sleeve	1	Amazon	B07TCDTFL2	Alex Tech 25ft – 1/2" Loom	9.5	9.5
25	Insulating material for inside oven gaps	1	Amazon	B00GT5Q6X0	CM-Ceramics 31" X 24" X 1" 2400 F	$33.50	$33.50
26	Fan for Cooling R-Pi and Housing Case	1	Amazon	B00N1Y4RLU	Gdstime 80mm x 80mm x 10mm DC 5V	8.99	8.99
						Blue Method	$465.28

Reflections on lessons, skills, and engineering practices learned

Redundant Design

We found that in the absence of some information, we benefited a lot from a design that was compatible with all expected cases. Without the SSRs being delivered during the PCB design phase, we decided it was best to include traces and througholes and pads for both potential outcomes we had defined based on our lack of information about the on resistance and other characteristics of the SSRs. Due to this decision we not only had the much more convenient to build direct controlled power to SSR option available when it turned out that was possible, we also now have two more controllable outputs the pcb can pass from the Raspberry Pi brain to future applications/improvements for our project, such as a servo motor door, and a more powerful cooling/convection fan.

Planning Ahead

We managed to complete our project on schedule in the summer quarter with 2 less total weeks available for normal capstone projects. We believe we benefited from our planning our milestones and tracking how tasks contributed to each milestone well over a month in advance, so when the final few weeks came we were not scrambling to get done additional unforeseen tasks.

Reading Data Sheets

One thing we spent a great deal of time learning how to do and doing was reading individual data sheets for parts we were considering buying or had already purchased, in order to set up theoretical circuits in LTSpice simulations or for understandings that would lead to better informed design decisions. Early on, we ordered one or two parts which turned out not to have expected characteristics; a thermocouple with too low of a sample rate, and an SSR with an inadequate thermal derating curve. As we moved forward we scanned the data sheets carefully for the requirements of our project to ensure compatible and competent design, and shunned parts with little to no documentation. This also helped inform us about the importance of documenting our own process and device characteristics.

Defining Requirements

As we moved through our goals, it was vital to be able to take abstract or high-level goals for our project and convert them into functional and nonfunctional requirements with specificity. If a large task with high level goals could be broken down into smaller tasks with specific goals, not only was there a greater understanding of the time and complexity involved in achieving the goal, but also we ourselves gained a better understanding of our likely design decisions. Spending time doing requirements analysis also helped us keep from veering off track into discussions involving irrelevant metrics and goals.

Acknowledgments

- Allen, Scotty, director. *Inside a PCB Soldering Factory - in China*. *YouTube*, YouTube, 8 June 2019, www.youtube.com/watch?v=24ehoo6RX8w.

- Gajdusek, Josef. *Atx*, 29 Oct. 2014, blog.atx.name/chibios-powered-reflow-oven-with-web-interface/.

- Pitrich, Karl. “0xPIT/ReflowOvenController.” *GitHub*, 0xPIT, 27 Apr. 2020, https://www.github.com/0xPIT/reflowOvenController.

- Yu, Xueming, director. *DIY SMD Reflow Oven for PCB Soldering, Using Toaster Oven Without Conversion or Transformation*. *YouTube*, YouTube, 14 Aug. 2020, www.youtube.com/watch?v=sdKK8a5j8Rg.

- *Kanboard*, kanboard.org.

- *JLCPCB*, jlcpcb.com.

- Rick Cordray, UWB Faculty Advisor
  rcordray@uw.edu

- Arnold Berger, Industrial Sponsor
  aberger@uw.edu

Appendices

User Manual Document Link

https://docs.google.com/document/d/1qaB-4ctdAdrjnjdcnZzgHnnEqXG2TZMfli4T9JXhRB8/

Definitions

AC	Alternating Current
DC	Direct Current
EE	Electrical Engineering
CE	Computer Engineering
CSSE	Computer Science and Software Engineering
Hz	Hertz, a unit of frequency
IDE	Interactive Development Environment
Altium	Computer Aided Board Design
PCB	Printed Circuit Board
PID	Proportional Integral Derivative, a control loop algorithm
UI	User Interface
UWB	University of Washington Bothell
SSR	Solid State Relay

Code Github Link

https://github.com/bhu413/reflow

Total Cost of Project

https://docs.google.com/spreadsheets/d/1I4osj6ROzDvsNyjKvthKbDbdxSsbv0t2/

Kanboard Project Management Software

https://kanboard.org/

PCB Gerber Files Link

https://github.com/bhu413/reflow/blob/main/FinalReflowOvenController.zip

Bibliography

1. Riemersma, Thiadmer. "Reflow Soldering Profiles." Reflow Soldering Profiles, www.compuphase.com/electronics/reflowsolderprofiles.htm.
2. Putzig, Nathaniel. *Thermal Inertia SI Unit Proposal*, nathaniel.putzig.com/research/tiu.html.
3. Gajdusek, Josef. *Atx*, 29 Oct. 2014, blog.atx.name/chibios-powered-reflow-oven-with-web-interface/.