GNG1103

Deliverable H: Prototype III

CEEDling: 3D Printer Monitoring System

Submitted by

GNG 1103, Section A3, Group 1

Magdalena Richardson, 7231925

Qaiz Mohamed, 300072323

Amesh Roy, 300073404

Midas Chin, 300066761

Lucas Hubert, 8386891

November 21, 2019

University of Ottawa

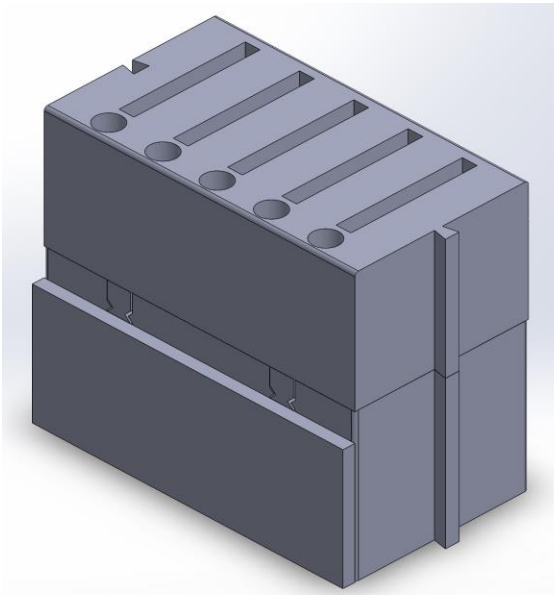
Introduction

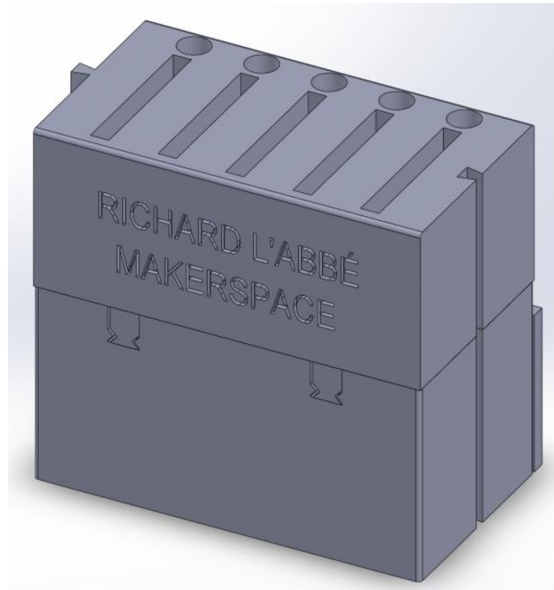
This document outlines the steps and tests involved in the last of our three prototypes that we created for this project. In this deliverable, we describe how we came to a functional prototype of our final design that incorporates all the subsystems of our product, including the electrical subsystem that includes all the sensing and circuitry, the software frontend subsystem that includes both UIs (one for CEED staff and one for CEED users), the software backend subsystem that includes the wireless communication protocol between our electrical system and our UI subsystem, and finally the hardware subsystem that incorporates the case for our SD card reader.

This third prototype builds on the previous two and incorporates all final changes and tests completed during the entire design process. An upgraded SD card holder that fits the SD card sockets as well as the LED lights is created for this prototype. We also refined our card sensing method and ran tests to verify if the NodeMCU was properly reading the cards and registering them as IN or OUT of their socket. We have also changed the way our UI operates and integrates information from the printers, moving away from an external database system to an all in one information management system. We also incorporated feedback from Ross Video and MakerSpace staff in our improved subsystems to ensure that our product reflects the client's needs as accurately as possible.

This prototype is one of the last building blocks leading up to Design Day and allows our team to further implement client feedback and tests into our design.

Hardware Subsystem





Figures 1 & 2 - Various views of the 3D model of the entire assembly made using Solidworks

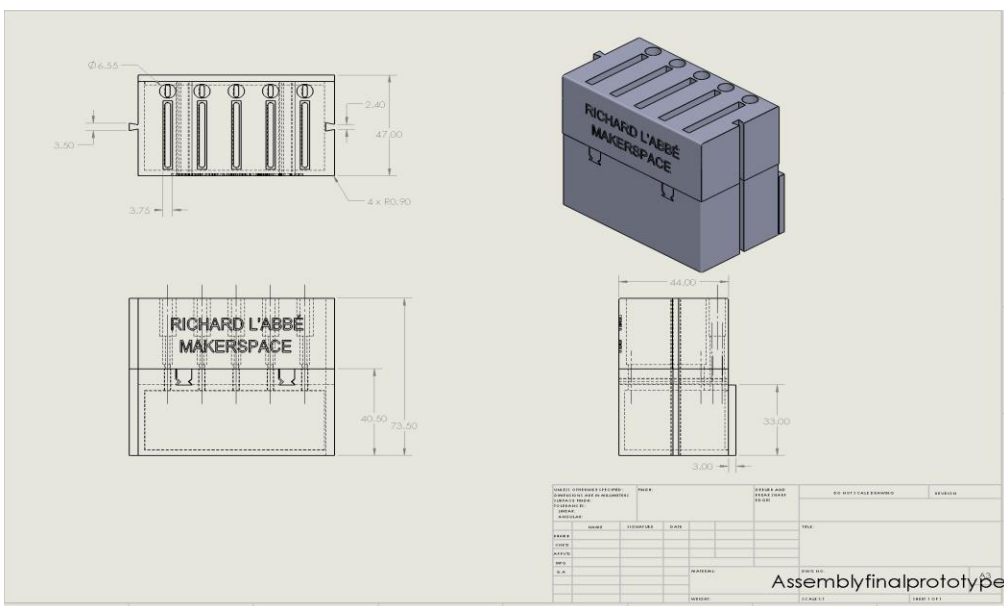


Figure 3 - Mechanical drawing of the assembly made using solidworks

Prototype three is the final SD card slot holder that is made up of three separate sub - components. It includes the SD card holder (top unit) that allows for the insertion of the SD card sockets as well as LED lights and is attached to the bottom unit using an upgraded interlocking system. The bottom sub - component houses the NodeMCU, multiplexer, as well as multiple wires that pass from the top unit to the bottom. Component three is a simple door that fits into the bottom sub - component to close it off and ensure the stability of the contents within. All components of this subsystem are 3D printed. Using the same cost per meter of PLA the total cost for this subsystem is $8.44 and is obtained using the following calculations:

Total length of PLA for 3 components = 5.06m + 6.13m + 1.04m = 12.23m
Cost per meter of PLA = $0.69/m
Therefore cost = ($0.69/m) (12.23m) = $8.44

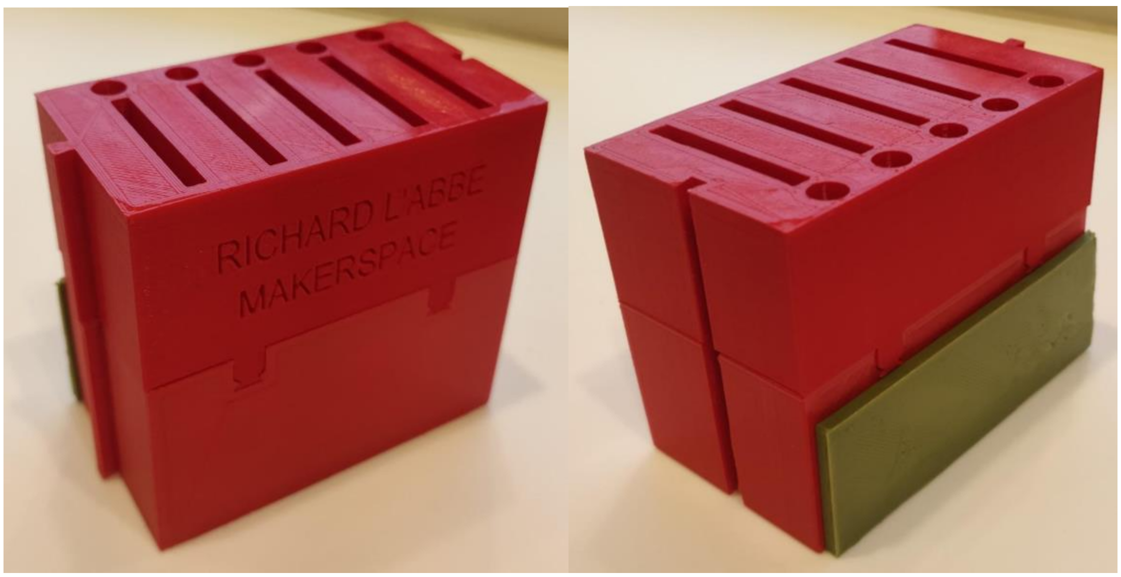
For this prototype the tests that will be carried out are the same tests performed on the previous prototype. These tests include verifying the integration with each subcomponent (including the upgraded interlocking system), checking to see whether all parts fit in as deemed necessary, and making sure that the electronic subsystem is integrated appropriately (spacing).

After printing all the subcomponents out and fixing them together, conclusive test results were established. In regards to the interlocking system between the top and bottom subcomponents, the tolerancing set is appropriate and allows for the top component to easily slide on to the bottom component. In terms of the door that will be attached to the bottom component, the size and shape are both suitable and lets a user place and remove it with ease. In addition, the thickness of the inner wall of the door is just the right size to avoid blocking the slots connecting the LED lights to the bottom component. After placing the appropriate parts where they belong, it can be seen visually that they fit in well and that no further changes are required. Furthermore, after inserting the ModeMCU, the wires, and the multiplexer within the bottom compartment, it can also be seen that all the parts fit in a firm and suitable manner. To conclude, all of the three main tests that were conducted on the previous prototype as well can be deemed successful.

As this is the final prototype, a few other important tests have also been carried out to investigate and verify more less - obvious properties of the model. The first test that was conducted was to investigate the strength and toughness of a previously 3D printed model. This property was tested using a drop test in the STEM building stairwell. After dropping the 3D printed model from the highest floor, then repeating the test with other models while changing the height of drop, it was concluded that the 3D model is able to withstand being dropped four floors and still display minimal plastic deformation. This is more than ideal, as the model would theoretically not suffer an impact as large as that when it is in use.

Another important addition to this final prototype, is the possibility of having a modular design where multiple SD slot holders and bottom holders can be added on. In the makerspace there are currently approximately thirty 3D printers that require the same amount of SD cards to be monitored. The current prototype is a small scale version that currently monitors only five cards but can be expanded to monitor as many cards as required. For this to be a possibility a slot and groove have been added to both the top and bottom components that would allow for identical sub - components to be easily added in a horizontal manner.

Although the final prototype is functional and meets the pre - established standards set by the team, there is still always room for improvement. One improvement for the current model that could possibly be implemented is to design an interlocking system that does not involve sliding. The reasoning behind this is because the current model allows for the top to easily slide off and therefore may accidentally slide off on impact. A possible solution to this is to design a snap system that allows the top component to be pushed into the bottom component and stay in place without coming out. This would allow for a sturdier and more reliable product. Another possible improvement is to change the door for the bottom component from a push door to one that would slide in place and stay there. Finally, the last improvement that can be made is to make the overall subsystem more compact in size. Currently, the prototype consists of one component on top of another which causes the overall look to be bulky. Going forward this can be changed by placing the components side by side and reducing the spacing between slots. In addition, the space required in the bottom compartment can also be reduced based on if the electronic subcomponent can be condensed/ compacted.



Figures 4 &5 - Multiple views of the final 3D printed SD card holder prototype.

Frontend Software Subsystem

Figures 5 & 6 - English and French View of Dashboard

The third prototype frontend software testing included new features including adding time left-in-day, English and French accessibility options, and linking two panels together to perform an employee and user kiosk set-up. The third prototype was a comprehensive prototype to test how different features interact on the main panel and on sub panels.

The first feature added was the panel linkage set-up to imitate a kiosk system where an employee has access to an edit pane. This simultaneously updates the user panel if changes are made. This feature was implemented in order to provide real-time updates to the database information and provide a nice one-way system between employee and user. Furthermore, it allows more features to be available on the employee side to modify the information being displayed. This feature was originally requested by CEED employees during client meetings. It was tested using two separate computers to view how quick the updates were between changes. It was noted that any updates happened effectively immediately and there was little to no end lag. However, the consequences have scalability was not tested within this time frame due to the lack of data points. This would be a good test in the future.

The second feature discussed was the time left-in-day feature. This was also requested by the CEED staff to help users predict if there will be enough time left in the day to finish their print. This was tested within the group to make sure it is clearly visible and functional.

The last feature that was implemented is the bilingual french/english option. The testing for this feature involved asking CEED users if 1). the UI was clear and well-understood, 2). the translation is accurate, 3). if it was easy to navigate. In general, it was shown that the employees and users could understand the UI and the translation was accurate. Furthermore, we asked a francophone to help proofread the translation.

Electrical Subsystem

The purpose of the third prototype electrical system was the following:

1. Finalize the circuit that will be used on Design Day.
2. Finalize the Arduino code to ensure that the LEDs and sockets work reliably.
3. Solder our components together.

Final Circuit

The purpose of this test was to ensure that we could fit all of our components on multiplexers[1] to keep our design modular, fitting in with our problem statement that our product could “adapt to a changing Makerspace”. We set a deadline of Nov. 21, 2019 as the last day to experiment with multiplexers, and while we were very successful in finding a way to get our inputs (our socket sensors) reading through the multiplexer, we were not able to incorporate our outputs (status LEDs) on the multiplexer. As we can see from below in Fig. 7, in the end we decided to keep the LEDs of our circuit as data pins on the NodeMCU, which is a change from our original plan of having all of our components put on multiplexers. Would we pick up this project and continue in the future, this would be one of the first things we would change; this would make our design scalable to all the SD-card-using devices in the Makerspace.

There were many challenges in this test and they were almost exclusively due to NodeMCU connectivity issues. We intend to catalogue these issues and the fixes that we found through trial and error for our final instruction manual in hopes that anyone taking on a similar project in the future will be able to find some much-needed support.

[1] Multiplexer here refers to the SN74HC165N shift-bit register - while technically not a multiplexer, it theoretically allows one to functionally expand the number data pins on a NodeMCU like a multiplexer.

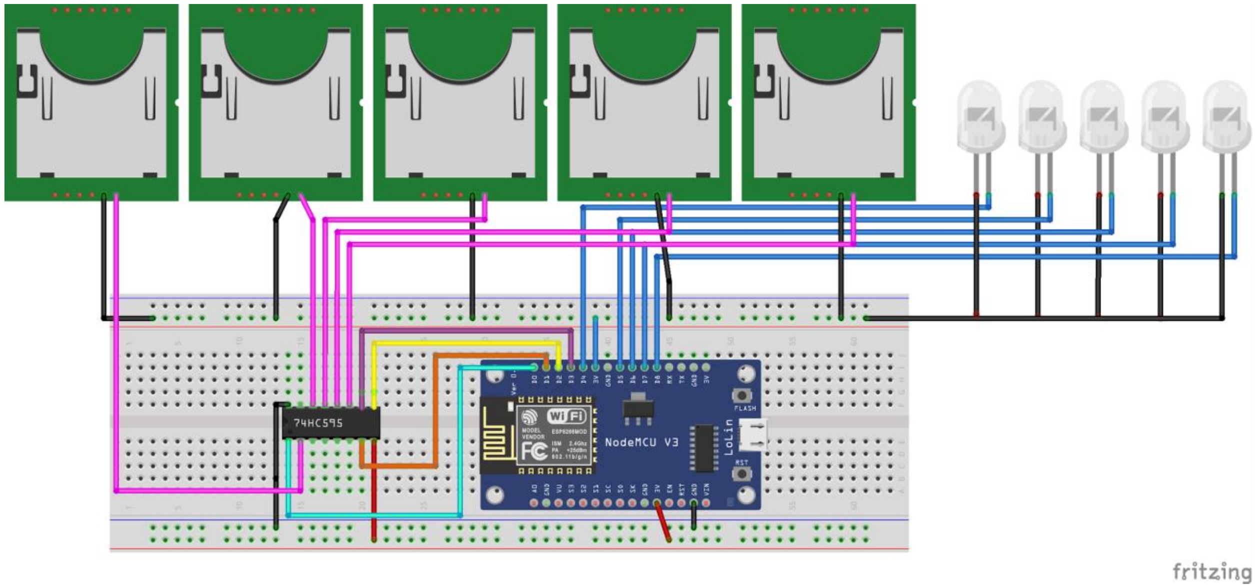


Figure 7 - The finalized circuit puts our SD card sockets on the multiplexer, but keeps our LEDs attached to the spare data pins of the NodeMCU.

Final Arduino Code

The purpose of this test was to see if our code was scalable to all five sockets of our final Design Day demo device. To do this, all five sockets were connected to the breadboard circuit and systematically tested with a corresponding LED that would turn green if the socket read as 'available', meaning a card was in the socket. We used two SD cards and clicked them into our holder in a variety of combinations and time delays and found that our program was able to detect changes to the card readers every time.

One thing that we noted was that there is a time delay of up to a second every time we would take a card out or put a card in. We do not foresee this as being a practical problem in the Makerspace settings as people are generally willing to wait a second for feedback, especially for a system when there is a significantly larger time delay between printer turnover.

Soldering

The purpose of this test was to see if our components could withstand the heat of a soldering iron and to see if all the components, once put together, could fit inside of our hardware container. As this was the last and most crucial step for our demo on design day, we decided to take this step very slowly and to test each component as it was placed in the circuit to ensure correct wiring.

This test started very well as all 5 LEDs and all 5 sockets were soldered with wire from the Makerspace, however we learned that the rigidity of the wire was too rigid for the fragile contacts of the sockets and every time a wire was moved too forcefully, it would break the contact off of the socket. Eventually all five sockets and LEDs were placed inside the case and were tested with the circuitry of the breadboard; the result was positive, although during subsequent steps, a socket occasionally had to be taken out and repaired. We determined that we could hot-glue the wires inside of the holder to make sure they didn't snap off of the socket.

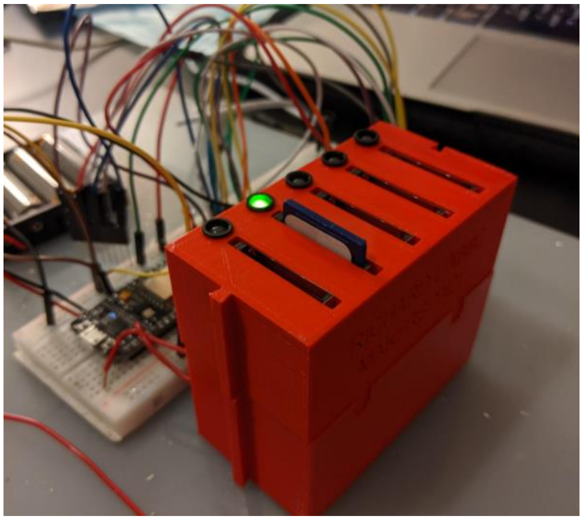


Figure 8 - An example of a component test, where the components in the holder were tested with the circuitry on the breadboard.

At some point during the soldering process, we tried to speed up and neglected to conduct small tests after each component was put down, and eventually we hit a point where the electrical system was not functional due to badly planned connections involving the power system that were creating too many shorts on the board. This resulted in having to re-solder one of the boards and basically start from square 2, picking up from the point where the components had been placed in the holder. This was frustrating, however the things that we learned from all the points leading up to this meant that the board had already been planned out and all that we needed to do was redo the connections. This took significantly less time to do the second time.

At the time of writing, there are two protoboards that contain all the components - one protoboard for the NodeMCU and header pins[2] and one board that contains our multiplexer, LED connections, and power wiring. At the moment, they do not both fit in the holder at the same time - this is largely due to the rigid wires that we used to connect our components.

If we were to continue this project in the future, we would use much smaller and more flexible wires and probably use a gentle epoxy to keep our components in our holder.

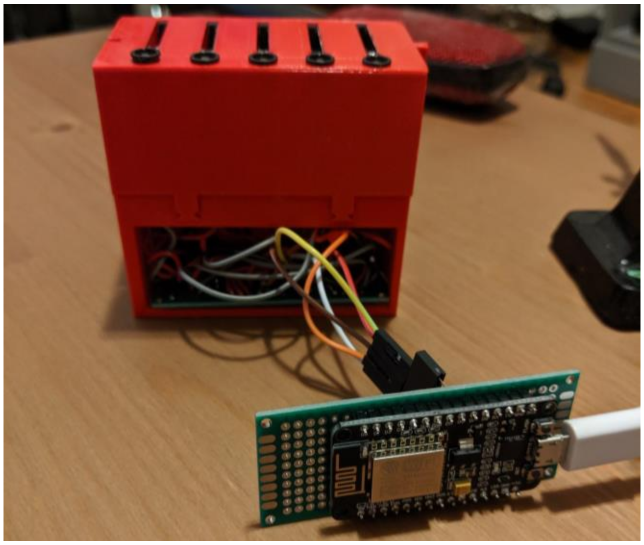


Figure 9 - At the time of writing, both boards do not fit in the holder due to rigid wiring.

[2] Header pins are required on this protoboard as the NodeMCU will prevent certain laptops from connecting to it if there are any data pins connected to the board when the upload starts. The header pins let us manually reset if we need to update the code.

Backend Software Subsystem

The third prototype testing for the backend software system included the incorporation of reading the push push card holder, determining the style and variable type in which information is going to be sent and converting that variable into information that makes sense to the user. At first, we played with the idea of sending the data in the form of a string where the string would look something like “010021031041”. Where each printer would occupy 3 characters. The first 2 characters being the printer number (01) and the third representing the status of the printer. The possible values for this status are 1 and 0 where 1 means the printer is available and 0 means the printer is in use. After performing some tests, we realized that we could streamline the process to a string that would look something like “01101” because when the values were being sent in a string, the status for printer 1 would be the first character of the string, the status for printer 2 would be the second character and so on. This means that we would not have to check for the printer number and the assumption could be made that character number would correspond to the printer number and we would only have to check for the status. After some more thought, to simplify the code, we decided that the final formatting would be in the form of an array with just the printer statuses ([0,1,1,0,1] for example). This way there would be no potential for string formatting issues. The incoming data is then parsed through, where the 1’s are converted to strings of the form “Available” and the 0’s are converted to strings of the form “In use”. These strings are then passed to the global setParams function which updates the status of printers on the UI. Shortly after testing this, we realized that the “In use” and “Available” also needed to be “Utilisé” and “Disponible” for the french version of the UI. To check for this, we simply used an if statement to determine whether or not one of the french parameters is active. If this is true, then the UI is in its french state , and the parameters could be set using the french words. If the condition is false, then the UI is in its english state and the english parameters can be used.

Despite the functionality of this code in order to make it easier to read and use for everyone, the code could be commented with explanations of what lines of code do. This would help to troubleshoot and resolve potential issues.

Conclusion

We have now brought all of our components together in a finalized prototype. While we are certain that we will tweak this system leading up to Design Day, we would be proud to present what we have to judges.

Our next steps leading up to Design Day will be to continue testing and tweaking so that our prototype will be in top form for when we present to our judges. We also intend to begin writing a detailed instruction manual of our project in case any team would like to do something similar in the future.

Above all else, this prototype has taught us the importance of communication and trust when coordinating efforts in a team, especially on a sophisticated project where everyone's efforts rely on everyone else's efforts. We have learned that it is easier to troubleshoot when we work in pairs, especially when blending components. We have also learned that despite careful planning, there will always be something unexpected that takes longer than planned to diagnose and overcome.