Introduction:

Kerbal Space Program is a simulator/game that uses real-life physics to simulate rockets and planes. The player can build their own rockets and fly them to space. Once in space the player can explore the solar system. The main feature of the game is that the physics is accurate to real life, simply flying in a straight line towards the destination, as often portrayed by sci-fi films, wouldn’t work. The player has must use their knowledge of orbital mechanics to plan the rocket correct maneuver so that they can reach their destination. While this may sound boring and dry, a large part of the game’s fun comes from the many, many explosions that the player will inevitably encounter.



The project aims to create a controller made of hardware that, through a software link, can control the rockets in game. A custom-built collection of switches, dials, displays and joysticks will provide a great increase to the game’s immersion and ease of control. A joystick becomes especially useful when controlling aircraft since a normal keyboard is not able to produce gentle movements. The style of the finished product means it will fit right in with a row of 1960s rocket control panels, increasing immersion. The tactile feel of the selected components will definitely give the user an unexpected amount of satisfaction when using the finished product. Overall it will be a great addition to the experience of Kerbal Space Program.

Inspiration and research:

Our original inspiration came from [this](https://www.reddit.com/r/KerbalSpaceProgram/comments/glmmt2/my_kustom_ksp_kontrol_box_prototype/) post on the Kerbal Space Program subreddit, a place online where people discuss the game. We then searched for some more examples on google images, which gave us many ideas for what we could include in our product. We then found the r/KerbalControllers subreddit where people discuss the more technical side of building these controllers, including the hardware and software required. This helped us understand the basics of code required and different methods of construction. Ultimately, the source that gave the most help for the code is the official documents for the Arduino library that we used to communicate with the computer and the game.

Development:

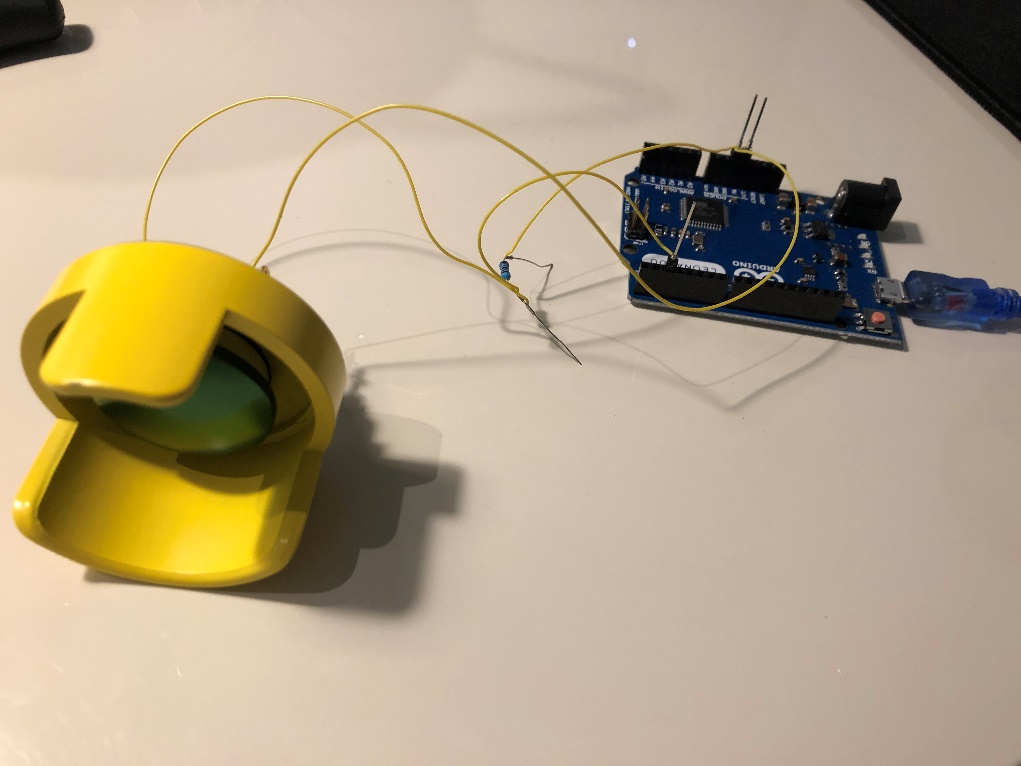
First day:

Today we started out with great hopes, but shortly after testing the very first prototype we hit a big hurdle. The Arduino simply wasn’t connecting to the game. After around 4 hours of troubleshooting, we were forced to give up and instead make the product in a different way. Rather than directly communicating with a game through a plugin, we will make the product emulate a keyboard. This can still satisfy most of the features, but things like 7 segment displays and analogue flight control will have to be left out. A few other features will also be challenging to emulate with a keyboard, but through some software trickery we should be able to make it work.

Second day:

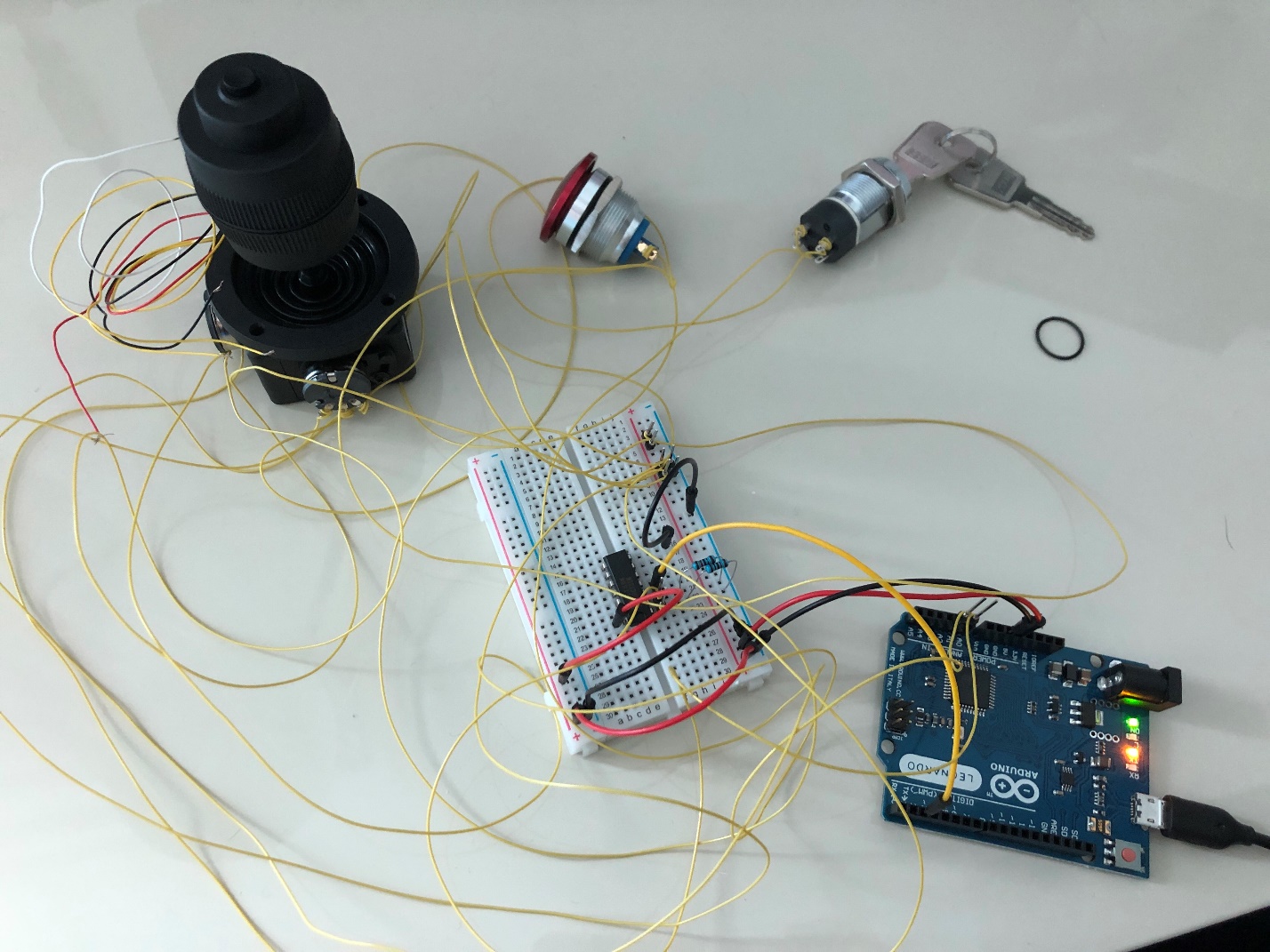
After more thinking, we’ve definitely decided to abandon the idea of communicating directly with the game through a plugin and a library. There are a few reasons for this, most obviously being we can’t get it to work. However, even if it does work, there are still a few problems and inconveniences with it. The plugin only supports an older version of the game, so the user will be required to go into the game manager and modify the game properties in order to revert to the older version. After that the user would also have to install a mod manager, install the plugin, and change the configuration files inside the plugin files. Overall, it’s just much more difficult to get it to work, and we don’t think it’s worth it for the few extra features it offers. We think the “plug and play” ability of using the controller to emulate a keyboard is much more important.

In order to get it to emulate a keyboard, a different microcontroller had to be used. I ordered it online and built a very simple prototype after it arrived. Making use of a few Arduino libraries, I was able to get it to work perfectly. I believe the rest of the construction design process will be relatively simple.



Third day:

Today we started development of a prototype that only includes a joystick and a stage button. We used wire wrapping as it is more secure than directly connecting to a breadboard but also faster and easier than soldering. The fundamental components of the code are working perfectly, and we were able to test the controller in game. It performed well and definitely added another layer of fun to the game.



Fourth day:

Today we finished the design and assembly of the prototype that includes one joystick and a stage button. There was some trouble with 3D printing, but the source of the problem was found and solved. Securing every part with screws was also a bit challenging as it takes quite a lot of force to thread the holes. In the end it works as intended, and really adds an extra layer of experience to the game.

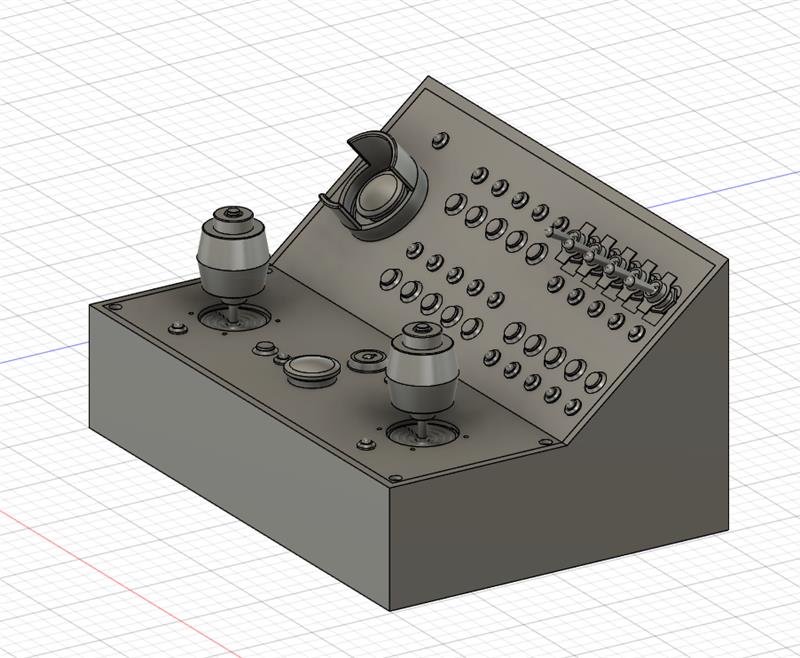


Fifth day:

Today we spent the morning designing the rough layout of the final product; we spent a lot of time, thinking of all different kinds of arrangements, and settled on a design with the main controls on a flat panel at the bottom, and all the other buttons and switches on a separate tilted panel. Afterwards we modeled the components in Fusion360 and started the design of the PCB. By the end of the day most components were modeled and a large part of the PCB was done.

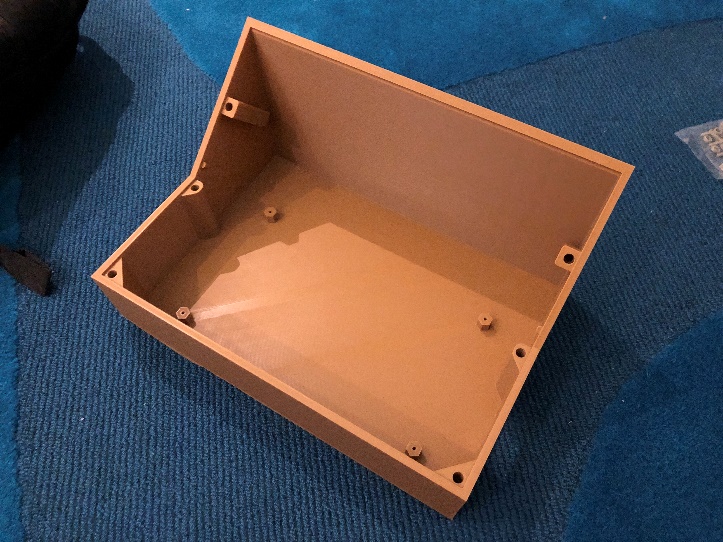
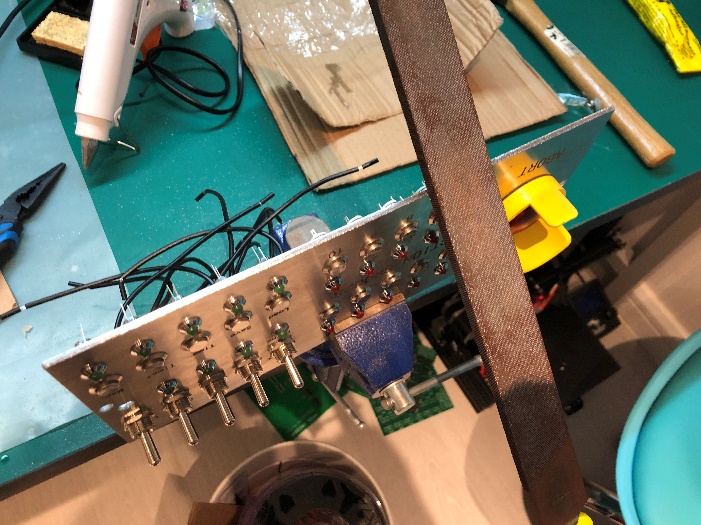
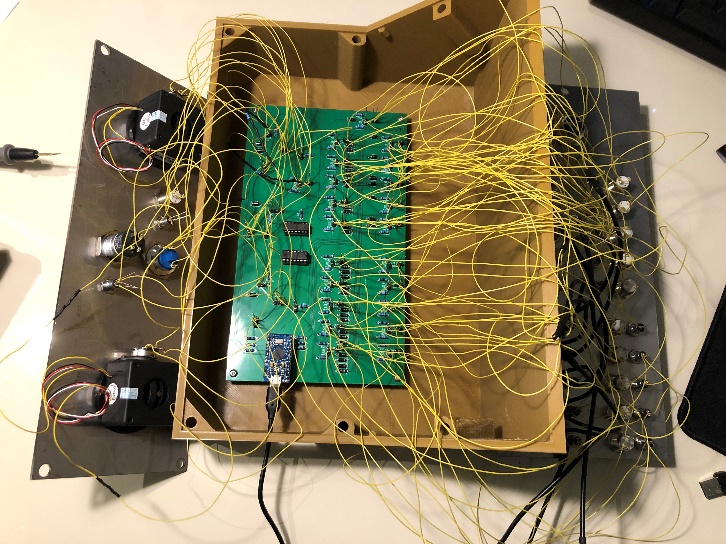
Sixth day:

Today we finished the design of the product in Fusion360 and the design of the PCB. The code is mostly complete but we still need to add a few lines to control the second joystick. With the completion of the design of the CAD model and PCB, we could order the laser-cut stainless-steel front panel as well as the circuit board. We had a bit of trouble sending the CAD file for the front panel to the manufacturer, but eventually it was solved. The only thing left to do is to brush up the code and wait for the front panel and PCB to arrive.



After the sixth day I did not record the specific things done in each day, so here are some pictures that roughly show what else I have done.





And here is the finished controller:

