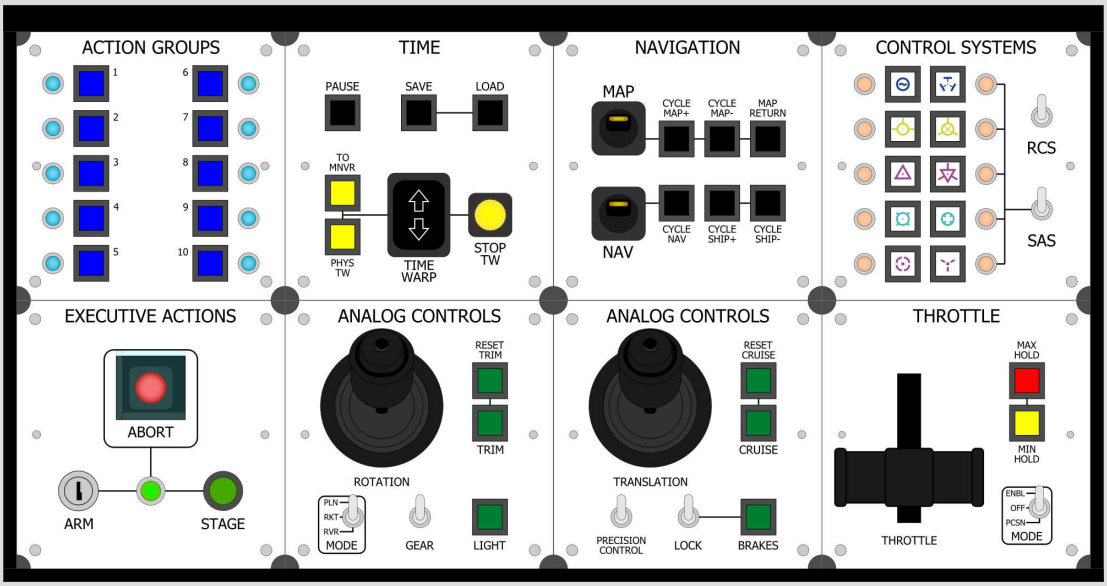
Untitled Space Craft

Troubleshooting Guide



Hopefully problems never happen, and when they do they can be frustrating. I'm happy to help fix any issues you may have with your USC controller. However, before contacting me, please make sure you've read this guide and have followed the steps where applicable. If you are still having issues after that, you may contact me directly.

A. The controller won't connect to KSP.

This is a common issue that could have many causes. Please check each of the following, no matter how obvious. Please note that the COM port might change throughout the tests.

- Check that the COM port is correct.
- Ensure that the power is plugged in.
- Make sure the controller is plugged directly into a USB port.
- Try a different USB port, perhaps swapping between USB3 or USB2.
- Make sure the LED is lighting up on the controller hub.
- Restart KSP.
- Unplug and replug both the power and USB on the controller.

If all those checks have passed and the controller still isn't connecting, then the next step is to try loading code onto it.

Please refer to the Code Flashing Guide and follow the instructions through the end.
https://drive.google.com/file/d/15OfTt_4WFQkeoATV0mpcZLOvZM7TydNE/view?usp=sharing

If that doesn't work, then please refer to part B.

B. Replacing the Arduino Nano/Micro

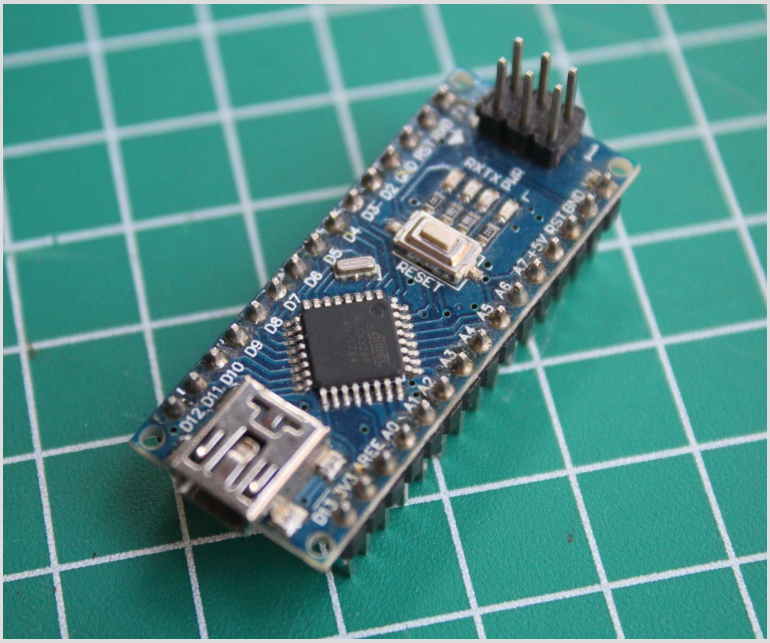
1. If the code failed to load, then it's a good bet that your arduino is fried. As a last resort you may try installing the CH341SER drivers and trying again.

https://drive.google.com/file/d/1L65JlEDR6Qxue4uuhsguE-bgvJnPHBzL/view?usp=sharing

2. Failing that, you will most likely require a new Arduino. Fortunately this is neither difficult nor expensive. You can acquire one from several online retailers such as Amazon or Alibaba for as low as a few dollars.

You should have already determined if you have a Nano or Micro when flashing the code, but if you aren't sure then you should confirm. The Serial Number on your hub should have a letter in it, series A, B, C, or D. Series B can only have Nano. Series C originally came with a Nano but can use Micro if you'd like to upgrade. Series D originally came with a Micro but can use Nano if you want to save some money. Arduino Micro is able to connect as a keyboard emulator. Sadly, this functionality isn't available on the Nano.

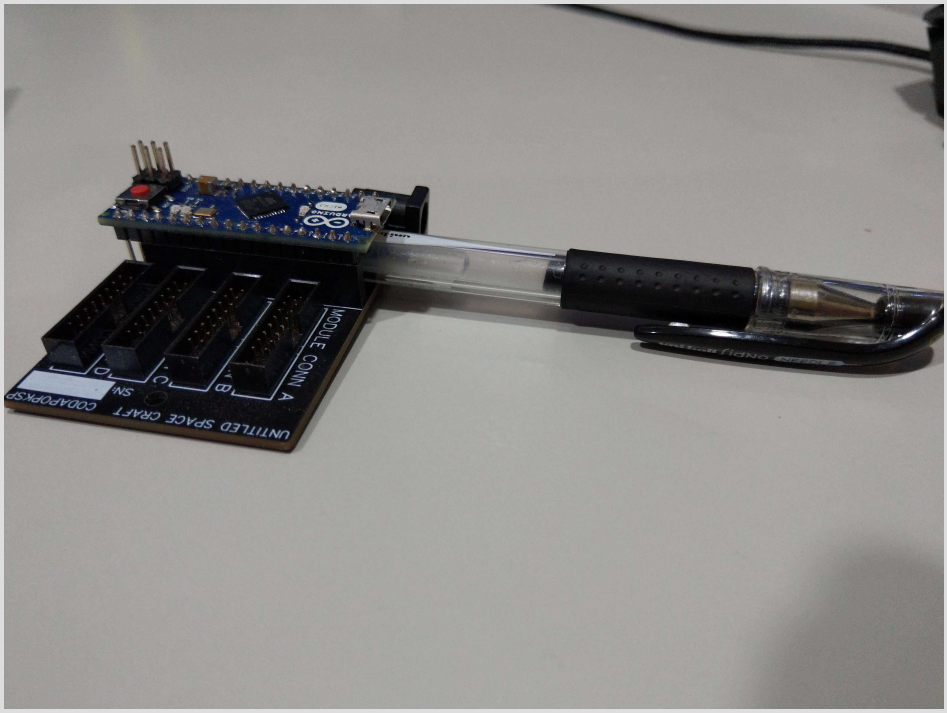
There are many different suppliers, but as long as you choose 'Arduino Nano' or 'Arduino Micro' then it should work. Just make sure that the USB port is the type that you want or are already using and that the board comes pre-soldered, like this picture.



3. Once you have a fresh Arduino, you may follow the previous steps to load the hub code onto it. Once that is complete, you will need to extract your old Arduino from your hub.

4. If your hub is already glued into your container, you will need to remove it, which might require breaking the glue seal. You may also be able to unscrew it.

5. Once you have the hub in hand, very gently wedge the old Arduino up by sticking a pen under it. Wedge up one side a tiny bit and then move to the other side. Alternate sides and lift it up little by little until it is free.



6. Once the old Arduino is out, you simply need to insert the new Arduino into the socket and press it down firmly. Make sure it is facing the right way! The USB port should be on the same side as the power jack.

At this point it is recommended that you try plugging in the modules and running the controller in the game to confirm it has been repaired before you glue it back together.

C. My analog sticks or throttle lever aren't working properly.

Analog sticks and the throttle lever are the only moving parts of the controller. They aren't descrete like computer code and thus need to be calibrated. This is typically already done for you, but sometimes they can shift over time or be pushed out of alignment during shipping.

Poorly-calibrated analog controls can manifest in a few different ways.

- The craft still has input in the direction of an analog stick even when you are not touching it, also called a center deadzone issue. (Be careful not to be fooled by SAS/Trim or other auto pilot mods.)
- The analog stick never reaches all the way in a given direction.
- The throttle still activates a little bit if even if the throttle is all the way down.
- The throttle never reaches 100% even when pushed all the way forward.

A lot of these problems can be solved by editing the deadzone calibration code in the settings.h file by following the Code Flashing Guide.
https://drive.google.com/file/d/15OfTt_4WFQkeoATV0mpcZLOvZM7TydNE/view?usp=sharing

The calibration values for the Throttle module are written on the PCB on the bottom of the module.

The analog sticks also have a physical calibration for two of the three axes which you can access underneath by loosening the screw next to the potentiometer and moving it around. It is not recommended that you try this without a multimeter and without at least a basic understanding of electronics.

If your analog stick or throttle is behaving erratically, then it's possible there is a deeper issue at play.

D. My controller connects but it disconnects frequently.

The controllers should be stable if set up properly, and I (Coda) can get many hours of play time out of mine without any issues. However, every system is different, and I can't claim to have tested them on every configuration. Currently, there are no known mods that conflict with the controllers, but it is something to keep in mind. It might be worth it to spin up a second build of KSP with only Simpit and its dependency mods to check for conflicts.

Controllers that consist of multiple containers (multiple USB cables) are known to have stability issues if connected using a USB hub. Thus, a solution here is to connect each container directly to a USB port.

An owner of a USC controller, Rettoph, has kindly created a more stable version of Simpit, which might allow for using USB hubs. At the moment it doesn't support Action Groups greater than 10. If you are having stability issues, then you can try downloading his build here: https://github.com/rettoph/KerbalSimpitRevamped/releases

However, you will need to use his Arduino library and reload the code to your hubs. You can find the arduino library packaged in the zip file, and you will need to install it manually. For Windows, typically it will need to be placed in Documents/Arduino/libraries. You will need to delete the KerbalSimpit folder there or else it may take priority. Then you will need to load the code by following the Code Flashing Guide: https://drive.google.com/file/d/15OfTt_4WFQkeoATV0mpcZLOvZM7TydNE/view?usp=sharing