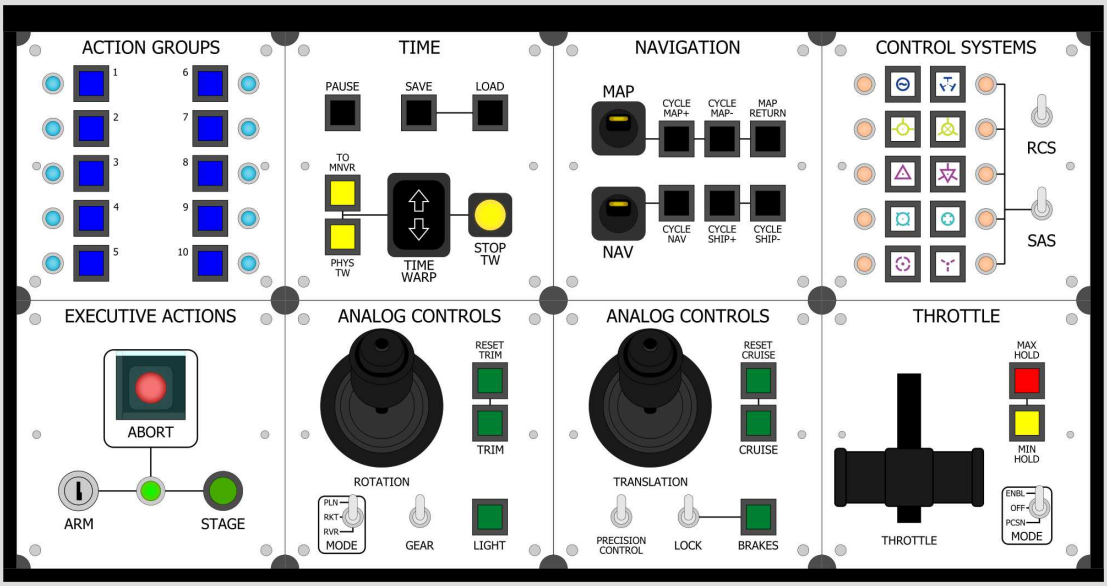
Untitled Space Craft

Keyboard Emulation Guide



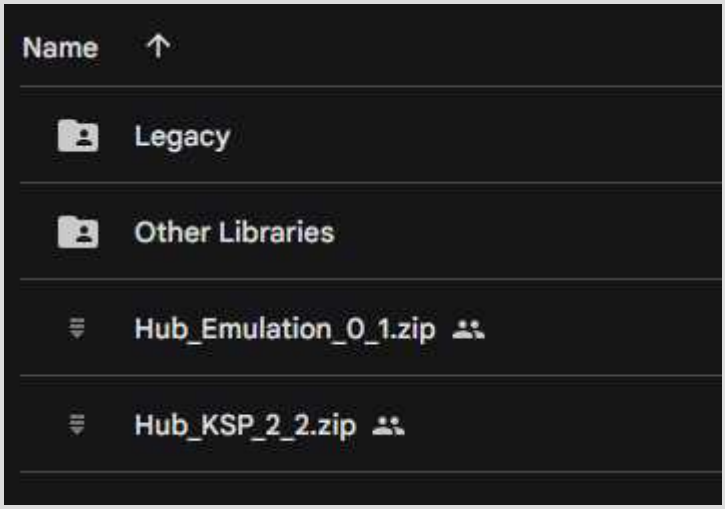
Keyboard Emulation Mode is a new Beta feature of USC controllers. It allows for the use of the controllers in other games via keyboard emulation. Using this mode means that the LEDs and displays in your controller will not work. It merely turns the controller into a mappable keyboard.

By contrast, KSP Mode interfaces directly with the game. You may return to KSP mode at any time to return functionality of the displays. *Tip: in KSP mode, you can unmap the keybinds for many of the keys in KSP to prevent accidentally hitting them on your keyboard.*

Keyboard Emulation Mode is only available for controllers that contain an Arduino Micro. If your controller arrived after May 2024, then your controller is likely powered with a Micro. You can check the serial number on your container hub to confirm. Series A and B cannot use Arduino Micro. Series C originally shipped with a Nano but can be upgraded with a Micro. Series D shipped with a Micro installed.

1. The bulk of this process involves flashing new firmware to the controller. You can find the Code Flashing Guide here: https://drive.google.com/file/d/15OfTt_4WFQkeoATV0mpcZLOvZM7TydNE/view?usp=sharing

The mode of your controller is dependent on which firmware is currently flashed to the board. Firmware titled "Hub_KSP_X_X" are for KSP Mode, while "Hub_Emulation_X_X" is for Emulation mode.



2. For KSP Mode, it should be ready to play after flashing. For Emulation Mode, you will need to map the buttons to your game or playstyle. This can be done in the file located in the Emulation code titled settings.h. Scroll down to find the section titled "Emulation Settings".

```
//|------------------------|
//|   Emulation Settings   |
//|------------------------|
  /*
    These values are the inputs for each button in the USC catalog.
    The values here are examples. To change the mapping,
    simply edit the value in the right-most column.
    It is recommended that you use lower case characters.

    You may find a list of modifiers and special keys here:
    https://www.arduino.cc/reference/en/language/functions/usb/keyboard/keyboardmodifiers/
  */

  // Module_InputName          |         Keyboard Value

  #define Action_Group_1                    '1'
  #define Action_Group_2                    '2'
  #define Action_Group_3                    '3'
  #define Action_Group_4                    '4'
  #define Action_Group_5                    '5'
  #define Action_Group_6                    '6'
  #define Action_Group_7                    '7'
  #define Action_Group_8                    '8'
  #define Action_Group_9                    '9'
  #define Action_Group_10                   '0'
  #define Action_Group_11                   '1'
  #define Action_Group_12                   '2'
  #define Action_Group_13                   '3'
  #define Action_Group_14                   '4'
  #define Action_Group_15                   '5'
  #define Action_Group_16                   '6'
  #define Action_Group_17                   '7'
  #define Action_Group_18                   '8'
```

In this section, you can locate all of the inputs available on the controllers. This includes modules that you might not have, which can be ignored.

Locate the section of a module that you wish to edit, and change the Keyboard Value to the desired input. You can accomplish this by putting the value in single quotes. For example, if you want the 1 button on the Action Groups module to input the letter x, you can change the '1' next to "#define Action_Group_1" into a 'x'. It is recommended that you use lower case letters. Other keys, such as Shift and Ctrl can be found at https://www.arduino.cc/reference/en/language/functions/usb/keyboard/keyboardmodifiers/

When you are done you may upload the code!