

CMDR Clicker's

Elite Dangerous Target Script

User Reference Guide

(v4.2.1)

Contents

Introduction	3
Acknowledgements and credits	4
Script Package Contents	5
Installation	6
Features	7
Usage	9
Elite Dangerous Configuration	12
MapKey Profiles	13
Training Mode.....	15
Modifiers	16
Axis Curves and Curve Control	18
Joystick Controller.....	24
Throttle Control Lever.....	41
Throttle Control Base.....	50
TFRP Rudder Controller	62
Keyboard Key Binds	63
ED_UserSettings File	65
Joystick Map - FULL.....	67
Throttle Map - FULL	68
Joystick Map – BASIC	69
Throttle Map – BASIC.....	70
Keyboard Utilisation	71
Key Bind Sheet	72
HOTAS Switch and Button Reference - FULL.....	73
HOTAS Switch and Button Reference – BASIC	74
Disclaimer	75

Introduction

PURPOSE:

My script package has been written for a dual purpose.

First, to enhance immersion and control within the game of Elite Dangerous.

Secondly, is to provide [TARGET](#) Script code examples to complement and extend beyond what is covered in the [Fast Script Basics User Manual](#) provided by Thrustmaster.

HISTORY:

Inspiration for this script started with [Aussiedoid's comprehensive Elite Dangerous script](#) (v3.1.0). I then created my own script to mimic my old Saitek X-55 basic configuration. I have since learned and used the advanced abilities of TARGET Script to create a "smart" configuration and have studied and lifted ideas and methods from Aussiedroid and others including Elite Dangerous Forum members, CMDRs Touille, Darkcyde and others.

PHILOSOPHY:

Whilst convenient, I tend not to use the [HOTAS](#) for non-HOTAS functions (eg I do not currently issue chat macros or fighter instructions etc via TARGET).

I've written the functions to suit my preferences, my play style and my hardware, however I have made a few concessions here to cater to users who may not have an EDTracker or rudder pedals.

My HARDWARE:

The script and supporting files are written and maintained to work with my current hardware setup. This includes a [Thrustmaster Warthog HOTAS](#), [Thrustmaster TFRP Pedals](#) and an [EDTracker Pro](#) (wireless). Keyboard and mouse usage is very limited but still required for certain functions. As it is, my script potentially does not suit playing with a [VR Headset](#).

COMPATIBILITY:

This script has been designed and tested to work with version 3.7 of [Elite Dangerous Horizons](#)/Fleet Carriers. This script will also work with the non-Horizons version of the game, however, quite obviously, Horizons based features then become somewhat redundant.

I have supplied a 'full' [bind file](#) which will work with or without the pedals, and an additional bind file which excludes the ED Tracker bindings. These should just work 'out of the box'.

NOTE:

The script uses extensive state tracking via the [status.json](#) journal file which is created and updated whilst the game is running. If this file becomes unreadable or unavailable, the script will abort.

The script requires Thrustmaster's TARGET Script editor ([v3.0.20.427](#)) to load, edit, compile and run and the formatting used suits the GUI console, not the script editor. Therefore, if you intend to edit this software, be careful changing the printf() formatting as output from this script displays differently due to font differences between the two.

I hope you enjoy this script and that it suits your requirements and playstyle. I look forward to any feedback and suggestions you may have for improvement.

Fly dangerously commanders!

[CMDR Clicker](#) o7

Acknowledgements and credits

Elite Dangerous is a copyright of Frontier Developments plc

T.A.R.G.E.T. and Thrustmaster are copyrights of Guillemot Corporation S.A.

Fast Script Basics User Manual by Thrustmaster

Original script inspiration and functionality by Aussiedroid.

<https://forums.frontier.co.uk/members/aussiedroid.21601/>

<https://forums.frontier.co.uk/threads/aussiedroids-enhanced-thrustmaster-warthog-script.293027/>

<https://github.com/Aussiedroid/AD-EDWarthogEnhancedScript>

Contribution and ideas lifted from Touille and Darkcyde.

<https://forums.frontier.co.uk/members/cmdr-touille.110127/>

<https://github.com/Touille/ED-Warthog-Target-Script>

<https://forums.frontier.co.uk/members/darkcyde.26482/>

Modified target.tmh (if included) was sourced from Sedenion.

<https://forums.eagle.ru/showthread.php?t=171098>

Analogue Slew Controller hardware upgrade – get rid of that crappy ‘mouse nub’ on the Throttle!

<https://deltasimelectronics.com/products/thumbstick-slew-sensor-adapter>

Last but not least, a lot of my learning came from assistance, directly or indirectly by several key people in particular;

On HOTAS/HOSAS/SIMPIT Discord...

@sYfte

@hon0

@dar|{cyde

Over on DCS World forums <https://forums.eagle.ru/> ...

@ivanwfr <https://forums.eagle.ru/member.php?u=82172>

@Sgt Coyle <https://forums.eagle.ru/member.php?u=88210>

@Drakoz <https://forums.eagle.ru/member.php?u=108387>

If you like my work and wish to contribute a small donation, please consider spotting me the price of a coffee!

<https://www.patreon.com/user?u=36436459>

Proceeds will go toward acquiring a T16000 TWCS so I can create a version of this script for the T16000 crowd.

Script Package Contents

When publishing or sharing this script the following should be included in the zipped package;

- The readme file
- The license file
- This reference manual
- ED_Main.tmc, ED_Functions.tmh, ED_Macros.ttm, ED_Toggles.tmh, ED_StateTracker.tmh,
- ED_MapKeyAssignment.tmh, ED_Defines.ttm, ED_GlobalVars.tmh, ED_UserSettings.tmh
- Voice.exe (users can find this on github and check out the source if they wish)
- sounder.exe (users can find this on github via the same author as voice.exe)
- Windows batch file (SAMPLE_Launcher.cmd) to use to launch Elite Dangerous, my TARGET script and supporting apps)
- The Elite Dangerous .bind files created to support this script
- The image files created to summarise the Joystick and Throttle button assignments
- The image files created to summarise keyboard key utilisation
- The change log for this version

Installation

- a. Copy the correct .bind file in the package's 'Bindings' folder to your Elite Dangerous bind file folder. This is usually at ...
"c:\users\%username%\AppData\Local\Frontier Developments\Elite Dangerous\Options\Bindings"
As 'AppData' can usually be hidden you may need to unhide it via the folder view options in Windows if navigating via the windows explorer. Or, you should be able to cut/paste the path above into a run box or address bar

NOTE:

- You should use the bind file in Bindings\Full folder within the zip package even if you don't have a TFRP Pedal controller
 - However, if you do not have an EDTracker use the bind file in the Bindings\No EDTracker\ folder
- b. Unzip the script files anywhere on your disk that you can easily find/get to
Example used in this document is c:\Thrustmaster\ED_TargetScript
Substitute your path within this document if you decide to save to somewhere else.
 - c. Copy voice.exe and sounder to c:\Thrustmaster\ED_TargetScript\ and set correct path in the ED_UserSettings.tmh file for both VoicePath and VoiceCMD variables
 - d. Copy Sounds folder to c:\Thrustmaster\ED_TargetScript\sounds\ and set correct path in ED_UserSettings file
 - e. Print out and study the image files provided in the 'Maps' folder.
 - f. Before you run the script open the ED_UserSettings.tmh file in notepad (I recommend using Notepad++) and ensure;
 - the path definitions for 'StatusFile', 'MyStatusFile', 'VoicePath' , 'SoundPath' and 'WAVPath' are all correct
 - Set 'DefaultVoice' to a Microsoft TTS voice installed on your PC
 - Set 'VoiceCMD' to include your preferred voice.exe command line switches
 - Set 'EnableTFRPRudder' to = 0 if you do not have this rudder controller
 - g. The Preferred method of running the script is via a Windows batch file.
ED_Launcher.cmd batch file is included in this package and is best copied to wherever you copied the scripts to in "step b" above.
I recommend you create a shortcut to this batch file on your desktop and run it as administrator. Please read the comments in this batch file for more information
 - h. The script MUST be running before you start the game.
 - i. When you run the game for the first time, go to Options | Controls and select the .bind file profile included in the zip package. (example: "Clicker-Warthog-v421")
 - j. If you do not have a TFRP Rudder controller you will need to correct the bindings for the following in game functions;
 - Ship yaw
 - SRV Steering
 - Camera yaw, FSS Camera yaw & SAA (DSS) Third-Person yaw
 - k. If you do not have an EDTracker, you will need to add bindings for Head look Pitch and Yaw if you intend to use this feature

Features

- 2 separate, user selectable MapKey profiles
 - FULL – all the bells and whistles
 - BASIC – no macros, no function calls, just send basic keystrokes
- 3 additional firing modes for primary and secondary triggers including
 - [Pulse Wave Scanner](#) mode (fires a pulse every 7 seconds)
 - [Discovery Scanner](#) mode (holds trigger for 6.1 seconds)
 - [Mining Laser](#) Mode (press to fire, press to stop)
- 5 user customisable, change on the fly curve profiles for Joystick, Rudder and Slew selectable via the autopilot switch
- [FA-OFF](#) automatically applies custom curves to joystick and rudder
- User customisable curve profiles for slew control and slider
- Slider curves (Radar tuning) automatically applied via flight mode tracking
- Advanced PIP Manager includes 6 PIP modes, all selectable on the fly
- Advanced Counter measures control includes single [SCB](#), SCB with auto [heatsink](#) and 'Double Bank' (2 x SCB + 1x heatsink) modes
- Silent Running 'Strobe' feature to toggle silent running on/off every 2 seconds
- Text-To-Speech engine provides additional voice feedback
 - turn on/off on the fly
 - volume control on the fly
 - includes a training mode which speaks each switch/button function
 - user selectable default voice
 - use different voices for different functions (in script only)
- Sound effects engine to add extra feedback in game
- Advanced state processor, which reads status.json twice per second and detects when key flags are changed.
- Additional flags tracked, saved to file and loaded when required, automatically
- Comprehensive print to console messages by most functions
- On the fly LED ON/OFF plus brightness control.
- Status LEDs accurately synchronised with game states for 5 key ship systems
- Flashing base LEDs on game start and flashing status LEDs on state change
- User Setting to enable or disable flashing LEDs
- Accurate game start and stop detection
- Macros included for;
 - Docking request with auto PIPs to shields
 - [Wing Beacon](#) On/Off
 - [Report Crimes](#) On/Off
 - Deploy and Recover the SRV
 - Dismiss/Recall Ship
 - [Station services](#) which can be set to auto on dock, or via switch
 - Launch+lift off+retract landing gear+25% throttle
 - Fast Game Mode switching (includes [anti-clog](#) code)
- Simple, logical target designator functions including system module targeting
- Speed brake function using timed landing gear deployment/retraction
- Advanced Super cruise / Hyper jump tracking with drop detection

(...features continued)

- Joystick, Throttle and Keyboard Utilisation images for both BASIC and FULL profiles
- Fully documented and commented code
- Comprehensive Key Bind Sheet file detailing standard and combo keyboard keys
- Toggle Engine and Weapon colours on the fly
- 2x comprehensive custom key bind files included
- Extensive use of layers using Throttle Pinky and/or Joystick Pinky switches
- Audio reminders when Throttle Pinky modifier switch is not in the middle
- Training Mode which prints to console and speaks the function of all buttons and switches
- ...and more!

Usage

All user configurable settings are in the ED_UserSettings.tmh file.
Avoid changing anything in ED_GlobalVars or ED_Defines files.

Feel free to use the ideas and methods for your own use in your own scripts and I encourage you to share these with other like-minded players.

This software uses the [MIT license](#), which can be found at the root of this package/repository.

Intent

Most buttons and switches do 'something' other than just send a simple keystroke or a [Direct-X](#) (DX#) command to the game.

Likewise, most of the axes available have different profiles (behaviours) depending on a user setting, or a specific combination of button presses, or, indeed something, that happens in game.

This guide is intended to describe what each button/switch does and how the different Joystick, Throttle, slider and Slew Control (mini stick) axes behave depending on situation or user definable configuration and in game actions.

This document is not intended to teach TARGET Script or walk you through my code. I will try to keep the scripting (code) description to an absolute minimum. Saying that, the script itself should provide to be an excellent resource in itself and provides many comments that can help you if you wish to learn TARGET Script, or modify mine to suit your needs.

Console

The script will output to the Target Script GUI console the result of many commands. Whilst not required to play the game, you can use this feature to ensure you are hitting the correct switch/button or sequence.

Likewise, the script will output error messages to the console and speech engine when you select an invalid button. Example: trying to lower the landing gear when in super cruise

(...Usage continued)

Text-To-Speech (TTS)

I have included a voice feedback feature that uses a Text-To-Speech engine and Microsoft's Voice capability to add more immersion and feedback when you select certain actions.

Many of the console messages are also converted to speech.

For this to work, copy **voice.exe** to your script folder and correctly set the path inside the ED_UserSettings.tmh file.

The volume for the Text-To-Speech function can be changed on the fly via FLAPU and FLAPD switch and the feature can be turned off completely via BSF (Boat Switch-Forward).

Refer to the individual switch references in the below chapters for more detail.

This Text-To-Speech feature is usually pretty straight forward...assuming Microsoft Speech services are enabled and you have at least one language pack installed (which you do, believe me!)

It can get a bit tricky ... **IF** ... you wish to use a voice not already installed on your PC.
More on that later.

I would suggest you first run voice.exe from the command line as follows:-

- 1) Open a command prompt at your script folder location
- 2) Type in 'voice -l' (lower case 'L') and hit enter

This will list any/all installed and available Microsoft TTS Voices installed on your PC.
If the list is empty, it means you do not have Microsoft Text To Speech feature enabled...refer to section below on customising voices.

Assuming you do have at least one voice installed you can test by typing in **voice "Hello"** and hit enter. This will say the text in the quotes using your PC's default voice.

Assuming the test above went well, let's open ED_UserSettings.tmh and go to the 'Text To Speech' section.

- a) Make sure 'EnableVoice' is set to 1 (or 'ENABLED')
- b) Make sure the 'VoicePath' alias is correct
- c) Make sure there is an alias for the voices installed in your PC (plenty of examples listed)
- d) Set 'DefaultVoice' to one of the installed aliases

Example: if the list you got shows 'Hazel', go ahead and set 'DefaultVoice = &Hazel;'

All going well, 'Hazel' will greet you when the game starts.

If you wish to use non-standard or not-installed voices here is a very basic rundown on what you need to do.

(...Usage – Text-To-Speech ... continued)

First, open ‘Speech Settings’ on your PC, select ‘Language’, then under ‘Preferred Languages’ add whichever languages you want to take advantage of.

Next, when prompted to confirm which features to install....make sure “speech” is enabled and go ahead and install.

Now, for the fun part (sic)...

By default, Microsoft Windows 10 does not enable all language voices to be accessible to all applications.

So, after installing the pack and restarting your PC, check if the TTS Voice is available via ‘**voice -l**’...if your new TTS voice shows up, you are good to go...if it does not show up in the list, what you gotta do is ... follow this web based tutorial for those voices you wish to make available to voice.exe...

<https://www.ghacks.net/2018/08/11/unlock-all-windows-10-tts-voices-system-wide-to-get-more-of-them/>

...it looks complicated at first, but is actually pretty simple...just take your time and in no time you’ll unlock the voice you want to use with this script.

Do not forget to go back, create an alias for the voice, and set ‘DefaultVoice’ accordingly.

This version of my script only uses one voice. I have included the ability to use different or multiple voices for future planned functionality.

Elite Dangerous Configuration

Additional configuration items I have set in game;

Options | Controls

(included in the bind files – here as a short reference for note)

- Throttle Axis Range = Forward Only
- Fwd Only Throttle Reverse Button Mode = Toggle
- Flight assist button mode = Toggle
- Rotational Correction = Toggle
- Firing Deploys Hard points = OFF
- Silent Running Button Mode = Toggle
- Cargo Scoop Button Mode = Toggle
- Microphone Mute Button Mode = Toggle
- Mute Button Mode = Toggle
- Microphone State Mode = Toggle
- Enable Context Menu = OFF
- UI Focus Mode = Cycle
- Autofocus on Text Input = OFF
- Looking at * Panel = Does Nothing
- Show CQC Score Screen Button Mode = Toggle
- Head look Button Mode = Toggle
- Centre when Head look Inactive = ON
- Head look Smoothing = ON
- Drive Assist Button Mode = Toggle
- Handbrake Button Mode = Toggle
- Drive Throttle Axis Range = Forward Only
- Drive Throttle Reverse Button Mode = Toggle
- Free Camera Throttle Axis Range = Forward Only
- DSS Front/Back Button Mode = Toggle

Options | Sound | Voice Volume | Ship Voice Volume (all 'ON' unless stated below)

- Power & Temperature
 - Silent Running Engaged = OFF
 - Silent Running Disengaged = OFF
- Landing & Docking
 - Landing Gear Deployed = OFF
 - Landing Gear Retracted = OFF

MapKey Profiles

There are two MapKey Profiles to choose from, FULL and BASIC, selectable in the ED_UserSettings file.

The **BASIC profile**, maps straight keyboard keypresses to the triggers, switches and buttons.

No macros, functions or modifiers are available in this profile so most of the features of the script are simply bypassed. This profile is for players looking for a straight WARTHOG profile and binding file combination, or players who are just starting out and want a simpler solution. The bind files, installation instructions etc are all compatible for both modes.

The **FULL profile** takes advantage of every feature, macro and function. It also has some rudimentary debug features, which can come in handy when you are not certain things are quite right.

Refer to

ED_UserSettings File chapter later in this guide.

Training Mode

Enable 'TrainingEnabled' in the ED_UserSettings file to take advantage of a training mode.

Enable 'AnnounceTraining' in ED_UserSettings file to announce via Voice feedback the selected Training mode status when the script starts or when the game stops.

When the game is not running, pressing any of the buttons or switch combinations described in the following chapters will result in both Voice feedback, as well as a console printout of the mapped function. Make sure you also check out the modified buttons and switches!

The training mode is customised for both FULL and BASIC profiles.

Modifiers

The 'FULL Profile' uses modifiers which do much the same thing as the 'SHIFT' or 'CTRL' keys do on a keyboard...they change or extend the use of the switch or button.

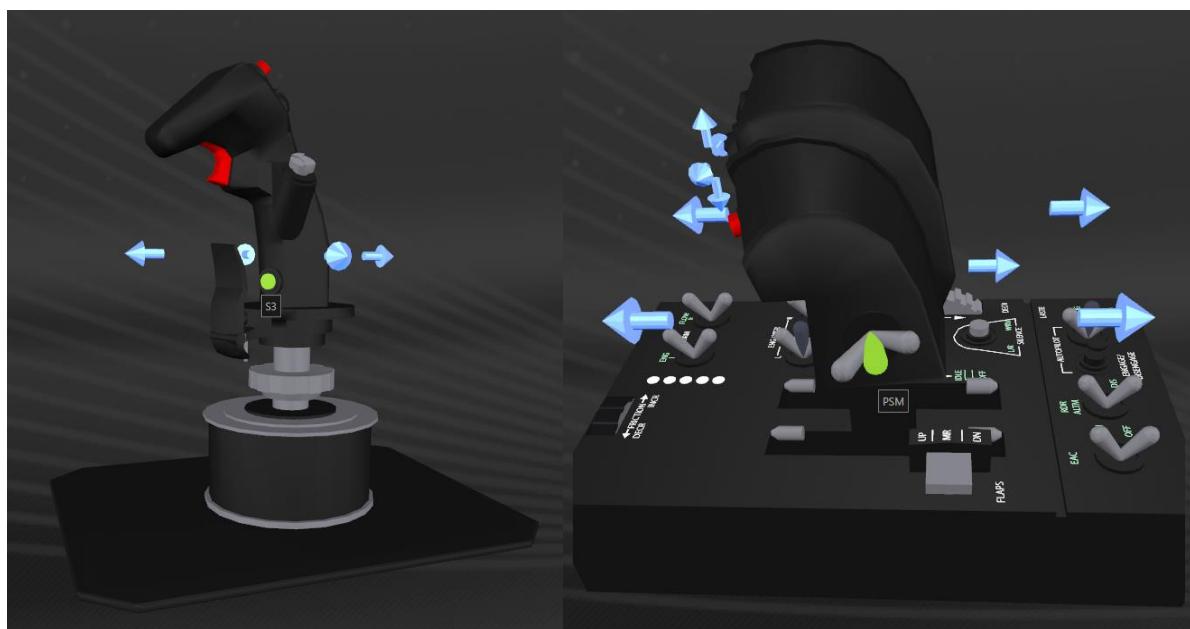
Using modifiers, each button or switch can have up to six different functions.

It is worth noting that I do not use modifiers on all switches and buttons ... only about half of the buttons I have mapped use modifiers.

The modifiers use the form I, O, U, M, D. The letters stand for 'In', 'Out', 'Up', 'Middle' and 'Down'

The script uses the Joystick S3 ('Nose wheel') button as the IO modifier. When pressed we are using the 'I' layer. 'O' layer is being used when it isn't pressed.

The script uses the Throttle PSF, PSM, PSB ('Pinky Switch') as the UMD modifier (layers). PSF is the 'U' layer, PSM is the 'M' layer and PSB is the 'D' layer (F=Forward, M=Middle, B=Back)



Joystick S3 'Nosewheel' switch

Throttle PSF/PSM/PSB 'Pinky' switch

In the ED_MapKeyAssignment.tmh file, you will see each switch and button mapped via a MapKey statement. This defines what we want to do when the switch is pressed.

It may look like one of the following four examples;

- MapKey(&Joystick, TG1, <action>);
No modifiers used. Press the trigger to do 'action'
- MapKeyIO(&Joystick, TG1, '<I' action>, '<O' action>);
Modifier 'IO' in play.
Press and hold S3, then the trigger to do 'action for I'
Press the trigger without pressing S3 to do 'action for O' (ie no modifier)

(...Modifiers continued)

- c. MapKeyUMD(&Joystick, TG1, <'U' action>, <'M' action>, <'D action'>);
Modifiers 'UMD' in play.
Flick Pinky Switch Forward (PSF) and press the trigger to do 'U action'
Leave or move Pinky Switch to Middle (PSM) and press the trigger to do 'M action'
Flick Pinky Switch Back (PSB) and press the trigger to do 'D action'

- d. MapKeyIOUMD(&Joystick, TG1, <IU>, <OU>, <IM>, <OM>, <ID>, <OD>);
The most complex mapping as you can see and has 6 separate actions depending on both S3 being pressed and/or position of the Throttle pinky switch!

In this user guide, I will list each switch position used in the script and if modified, will use the following example terminology;

<u>Switch</u>	<u>Usage</u>	<u>Means...</u>
TG1 – IU	S3+PSF+TG1	Press+hold S3 (I), Flick PSF (U), then press the main trigger
TG1 – OM	TG1	Just press the Joystick main trigger (no effective modifier) (S3 not pressed and Throttle Pinky Switch in the middle position)
S1 – U	PSF+S1	Flick PSF (U), then press the Joystick S1 ('Master Mode') button
H4P – D	PSB+H4P	Flick PSB (D), then press the Joystick CMS Switch
S4	S4	(no modifiers) ... just press the Joystick S4 switch (pinky lever)

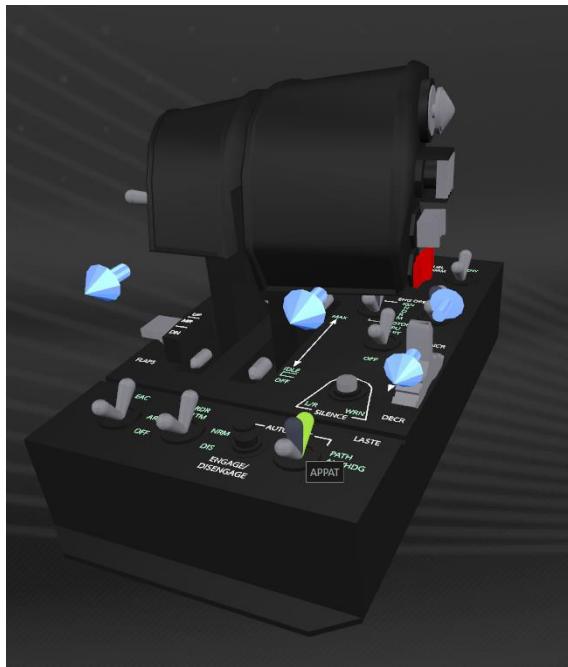
In other words, I will only describe the modifiers where they are used for that switch mapping in script.

I will then go on to describe the 'action' or function of each switch in use and I will include a reference picture of the switch being described.

To further complicate matters, the script does make use of the ability to detect when a switch/button is released in cases where I need this...I will avoid using these references here for simplicity. Refer to the MapKeyAssignment file and look for 'MapKeyR', 'MapKeyRIO', 'MapKeyRUMD' or 'MapKeyRIOUMD' statements.

Axis Curves and Curve Control

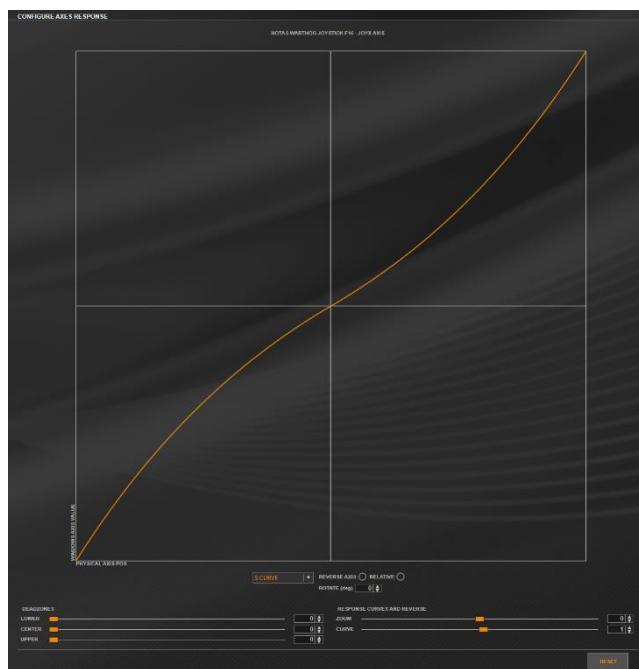
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APPAT-O	APPAT	Set Joystick and Rudder curves to MILD (1)
APPAT-I	S3+APPAT	Set Joystick and Rudder curves to MEDIUM (2)



Throttle APPAT (Autopilot Path)

MILD applies a small amount of curves applied to the joystick and rudder.
The joystick and rudder have a slight non-linear rate of response.

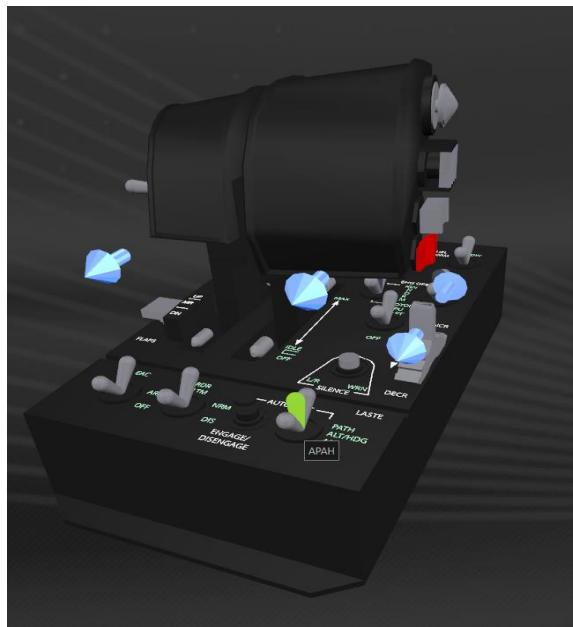
The S-Curve causes the response to be slightly slower at the beginning of travel away from centre and gets faster the further from centre you move the stick (or rudder).



Joystick Curves set to 1 (Non-Linear Response)

(...curves continued)

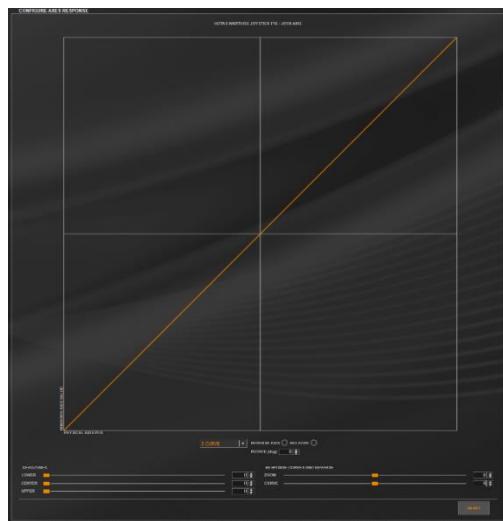
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APAH	APAH	Set Joystick and Rudder curves to OFF (0)



Throttle APAH (Autopilot ALT/HDG)

Default position for this switch. No curves means that the joystick and rudder axes have a linear response. That is, it increases the roll or pitch rate in a consistent manner as you move the stick all the way left, right, up, or down.

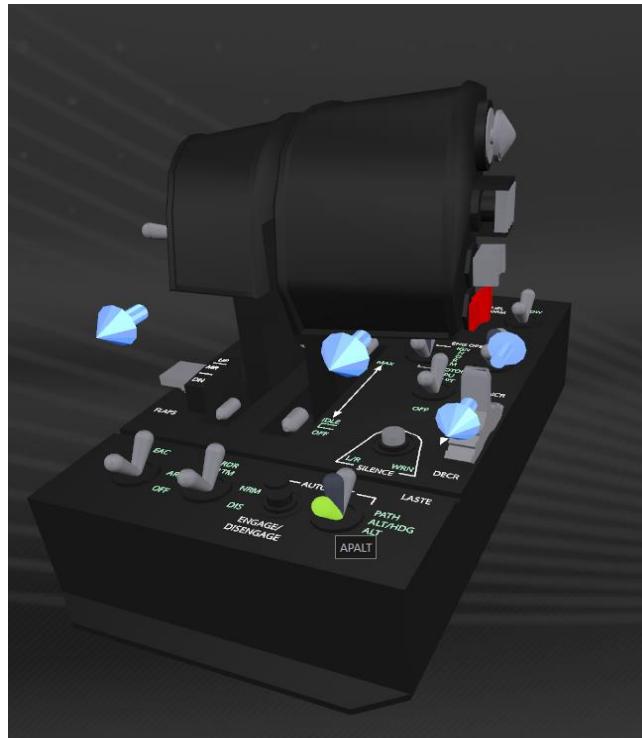
As the script applies the same curve profiles to the joystick and rudder when switching, the same theory applies to the rudder.



Joystick Curves set to 0 (linear response)

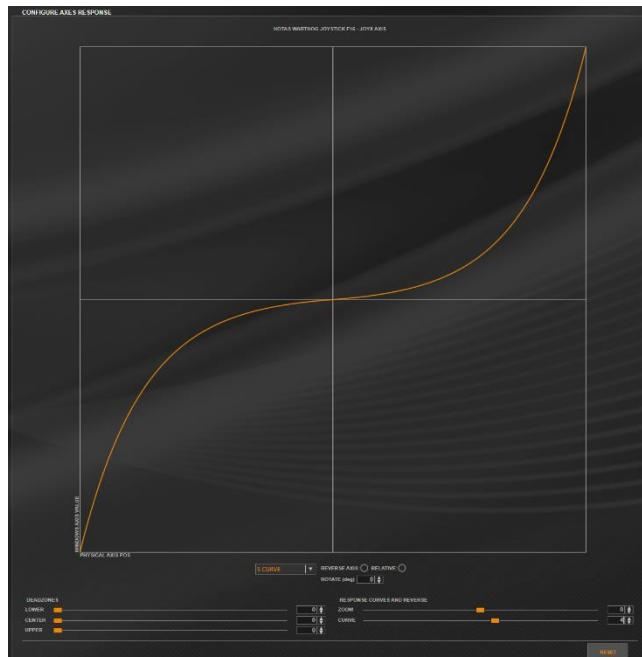
(...curves continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APALT-O	APALT	Set Joystick and Rudder curves to SLOW (3)
APALT-I	S3+APALT	Set Joystick and Rudder curves to SLOWEST (4)



Throttle APALT (Autopilot ALT)

At a curve profile of 'SLOW' (3) or 'SLOWEST' (4), the Joystick/Rudder response starts quite a bit slower and then gets a fair bit faster the more you move the throttle or rudder from the centre position. This is most useful when learning and using Flight Assist Off (FA-OFF).



Joystick Curves set to 4 (non-linear response)

(...curves continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
S4	S4	1 st press - Toggles FA-OFF and sets joystick & rudder curves to the value you choose in the UserSettings file (eg MEDIUM or SLOW) 2 nd press - Toggles FA-ON and resets curve profile to the associated autopilot switch setting



Joystick S4 (Pinky lever)

FA-OFF: Status LED 1

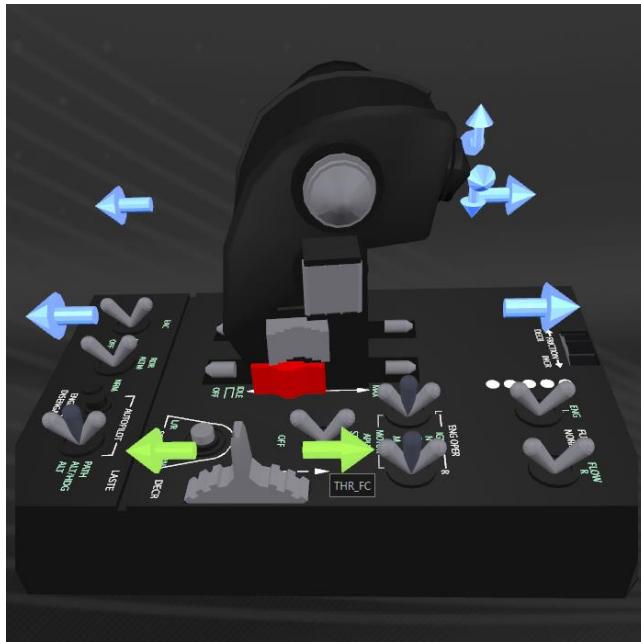
The curves that FA-OFF will use can be set via the 'FAOFFCurves' variable within the ED_UserSettings.tmh file. Valid settings are 'OFF', 'MILD', 'MEDIUM' or 'SLOW'

Console printout each time curves change can be enabled via 'CurveInfo' variable within ED_UserSettings.tmh file. Valid settings are 'PRINT' or 'NOPRINT'

Note: The above settings are case sensitive.

(...curves continued)

<u>Axis</u>	<u>Usage</u>	<u>Action</u>
THR_FC	THR_FC	Used to adjust radar range



Throttle THR_FC (Slider)

The slider axis requires 'SetCustomCurve' which is selected automatically depending on flight mode. Flight mode is detected in the script each time your flight mode changes from normal, super cruise, FSS Scanner or DSS Scanning (surface probes, front/rear planet view).

- Mode 0: linear. Used for FSS Mode Radar
Curve Profile 0 = (0,0, 25,25, 50,50, 75,75, 100,100)
- Mode 1: non-linear. Used for normal flight and SRV
Curve Profile 1 = (0,0, 40,25, 75,50, 85,75, 100,100)
i.e. When slider is at 40%, Windows DX value is at 25% etc.
- Mode 2: non-linear. Used for Super cruise
Curve profile 2 = (0,0, 50,10, 75,30, 90,50, 100,100)
- Note: J-Curve settings (via 'SetJCurve' statement in script) do not work as well as 'SetCustomCurve'

Most of the settings for each of the curve functions can be found within the ED_UserSettings.tmh file under the 'Axis Curves' section, or in the ED_Functions file (look for fnSetJoystickCurves() and fnSetSliderCurve() functions).

Refer 'Axes' chapter starting on p27 in the [Target Script Editor Basics Manual v1.5](#) for more details on 'S', 'J' and 'Custom' curves.

(...curves continued)

<u>Axis</u>	<u>Usage</u>	<u>Action</u>
SCX	SCX	Slew Control X-Axis. Used for Lateral Thrusters
SCY	SCY	Slew Control Y-Axis. Used for Vertical Thrusters



Throttle SCX (Slew Control)

Throttle SCY (Slew Control)

Curves for the Slew Control are also set when we change the Joystick curves.
The three profiles are all set the same (linear) but can be changed via curve arrays within ED_UserSettings.tmh.

I use a slight centre dead zone, as this control is quite sensitive.

NOTE: I have modified my WARTHOG HOTAS by replacing the slew control 'nub' with an analogue control replacement that I obtained from:

<https://deltasimelectronics.com/products/thumbstick-slew-sensor-adapter>

This is a fantastic upgrade to the HOTAS, which I heartily recommend.
It turns what I consider a relatively useless control that I did not/would not use, into something supremely useable which I cannot do without!

Joystick Controller

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
TG1-U	PSF+TG1	Increment Primary Trigger Action - Discovery Scanner Mode - Mining Laser Mode - Pulse Wave Scanner Mode
TG1-M	TG1	Primary trigger default action (press to fire, release to stop)
TG1-D	PSB+TG1	Disco scan. Press+Release to fire. Trigger released after 6.1 secs Mining laser. Press to fire, press to stop Pulse Wave Scanner. Press+Release to start. Pulse sent every 7 seconds. Press+Release to stop



Joystick TG1 (Primary Fire)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
TG2	TG2	(not mapped)
S1-U	PSF+S1	Toggle HUD between Analysis and Combat Modes
S1-M	S1	Toggle Hard points (normal flight mode) Toggle HUD Mode (in Super Cruise flight mode)
S1-D	PSB+S1	Toggle Joystick X-Axis between ROLL and YAW



Joystick S1 (Master Mode Control)

NOTE: When in Super cruise, S1-M toggles HUD mode automatically

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
S2-U	PSF+S2	Increment Secondary Trigger Action - Discovery Scanner Mode - Mining Laser Mode - Pulse Wave Scanner Mode
S2-M	S2	Secondary trigger default action (press to fire, release to stop)
S2-D	PSB+S2	Disco scan. Press+Release to fire. Trigger released after 6.1 secs Mining laser. Press to fire, press to stop Pulse Wave Scanner. Press+Release to start. Pulse sent every 7 seconds. Press+Release to stop



Joystick S2 (Secondary Fire/Weapons Release)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H1U-U	PSF+H1U	Toggle FSS Mode
H1U-M	H1U	Select Next Fire Group
H1U-D	PSB+H1U	Turn off DSS Mode



Joystick H1U (China Hat – Up)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H1D	H1D	Select Previous Fire Group



Joystick H1D (China Hat – Down)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H1L	H1L	Tempo. Press and hold for 0.5 seconds toggles landing gear Short press does nothing.



Joystick H1L (China Hat – Left)

Landing Gear: Status LED 5

Press and hold H1L for 0.5 Seconds to toggle the landing gear.

This prevents accidental deployment during combat if you accidentally hit H1L instead of H1U (Next Fire group) or H1D (Previous Fire group)

The 0.5-second delay can be changed via the 'LongPress' variable defined in the UserSettings file.

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H1R-U	PSF+H1R	Toggle Night Vision ON/OFF
H1R-M	H1R	Tempo. Press and hold for 0.5 seconds toggles lights Short press does nothing. Prevents accidental toggle of lights
H1R-D	PSB+H1R	(not mapped)



Joystick H1R (China Hat – Right)

Toggle Lights: Status LED 2

Ship Lights simply toggle ON/OFF

SRV lights cycle ON/HI BEAM/OFF

The 0.5-second delay can be changed via the 'LongPress' variable defined in the UserSettings file.

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H2U-U	PSF+H2U	Toggle Planet View, Front/Back (Must be in FSS Mode)
H2U-M	H2U	Select Target Ahead
H2U-D	PSB+H2U	Select Target Ahead



Joystick H2U (Target Designator – Up)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H2D-U	PSF+H2D	(not mapped)
H2D-M	H2D	Select Highest Threat
H2D-D	PSB+H2D	Select Wing Man's Target



Joystick H2D (Target Designator – Down)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H2L-U	PSF+H2L	(not mapped)
H2L-M	H2L	Select next ship
H2L-D	PSB+H2L	Select next wingman (sequences each wingman in turn)



Joystick H2L (Target Designator – Left)

(... JOYSTICK continued)

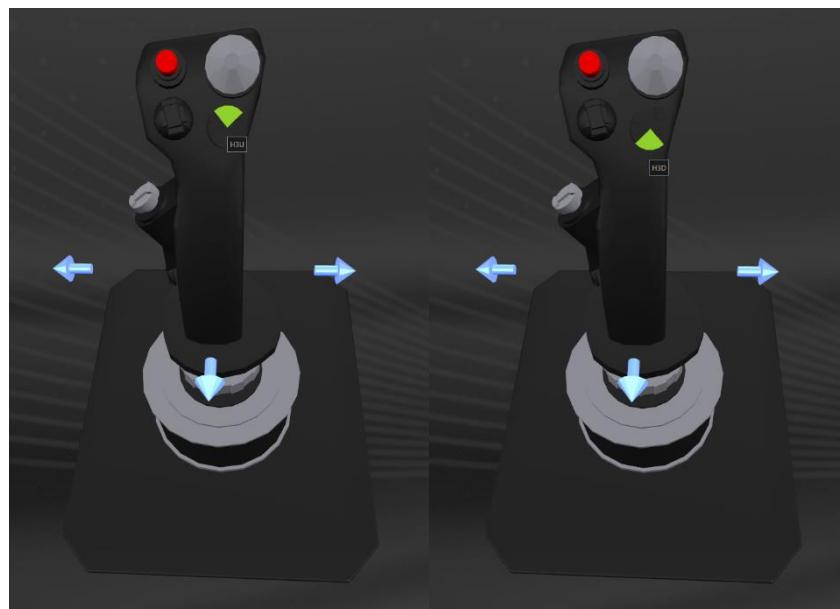
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H2R-U	PSF+H2R	(not mapped)
H2R-M	H2R	Select next hostile ship
H2R-D	PSB+H2R	Select Wingman's NAV Lock



Joystick H2R (Target Designator – Right)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H3U	H3U	PIP Management – PIPs to Engines
H3D	H3D	Balance PIPs
H3L	H3L	PIPs to Systems (Shields)
H3R	H3R	PIPs to Weapons



Joystick H3D (Data Mgmt – Up) Joystick H3U (Data Mgmt – Down)



Joystick H3L (Data Mgmt – Left) Joystick H3R (Data Mgmt – Right)

The script has six PIP Modes selectable via H4P (see below)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H4P-U	PSF+H4P	Reset PIP Mode to DefaultPIPMode (set in ED_UserSettings.tmh)
H4P-M	H4P	Select Next PIP Mode. (0 - 5) (5 wraps to 0)
H4P-D	PSB+H4P	Select Previous PIP Mode (5 - 0) (0 wraps to 5)



Joystick H4P (Countermeasures – Press)

PIP Mode 0:	Single PIPs	1 press = 1 PIP
PIP Mode 1:	Double PIPs	1 press = 2 PIPs
PIP Mode 2:	Attack	SYS = [4 0 2], ENG = [0 4 2], WEP = [0 2 4]
PIP Mode 3:	Defend	SYS = [4 2 0], ENG = [2 4 0], WEP = [2 0 4]
PIP Mode 4:	Recharge	SYS = [4 1 1], ENG = [1 4 1], WEP = [1 1 4]
PIP Mode 5:	3+3	SYS = [3 3 0], ENG = [0 3 3], WEP = [3 0 3]

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H4U-U	PSF+H4U	'Double Bank' Fire 2x Shield Cell Banks (SCB) + 1x Heatsink
H4U-M	H4U	Tempo. Short Press = Fire 1x SCB, Long Press = 1x SCB + 1x Heatsink
H4U-D	PSB+H4U	FSS or Camera Zoom - IN



Joystick H4U (Countermeasures – UP)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H4D-U	PSF+H4D	(not mapped)
H4D-M	H4D	Fire 1x Heatsink
H4D-D	PSB+H4D	FSS or Camera Zoom – OUT



Joystick H4D (Countermeasures – DOWN)

(... JOYSTICK continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H4L-U	PSF+H4L	(not mapped)
H4L-M	H4L	Electronic Countermeasures (ECM). Hold to charge, Release to Fire
H4L-D	PSB+H4L	Camera Blur – OUT



Joystick H4L (Countermeasures – LEFT)

(... JOYSTICK continued)

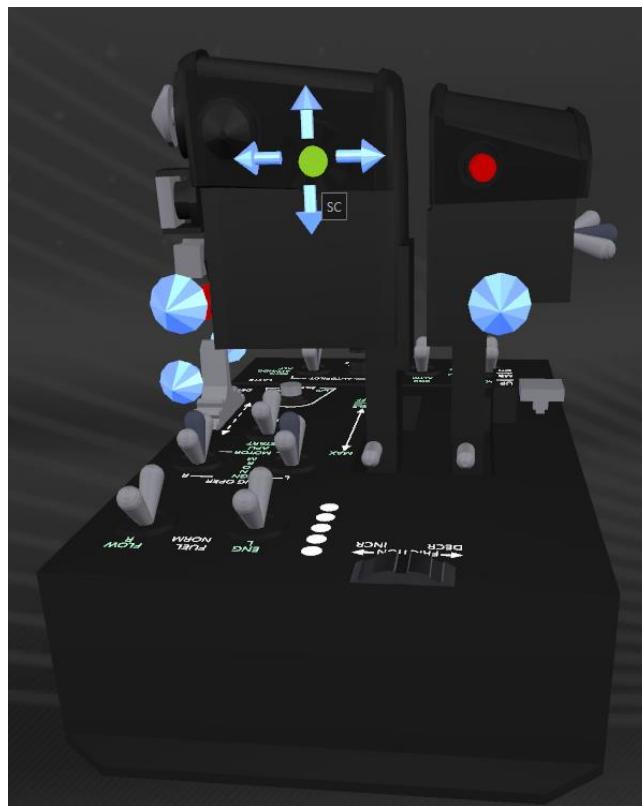
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
H4R-U	PSF+H4R	(not mapped)
H4R-M	H4R	Fire 1x Chaff
H4R-D	PSB+H4R	Camera Blur – IN



Joystick H4R (Countermeasures – RIGHT)

Throttle Control Lever

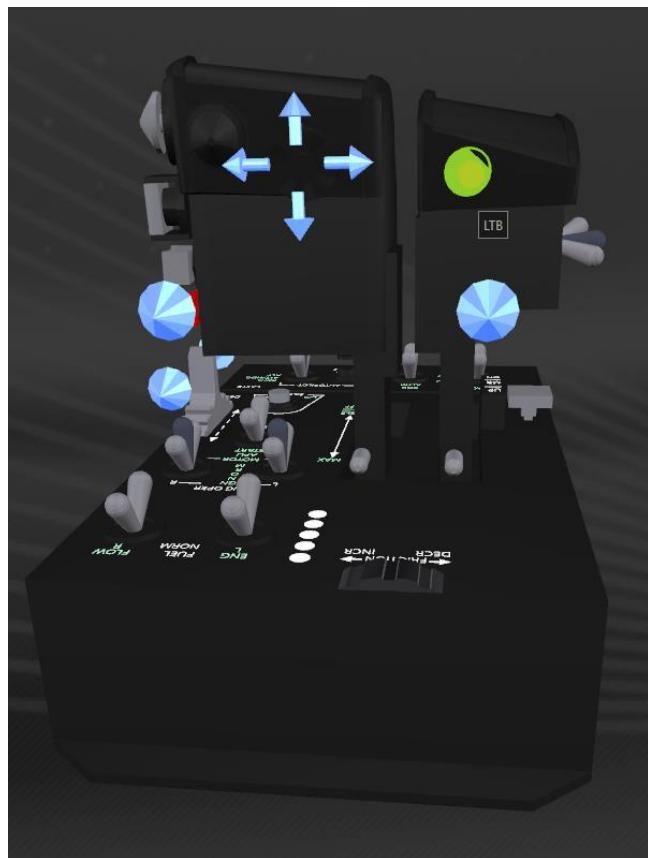
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
SC-U	PSF+SC	Toggle Connection status. Shows Bandwidth counter
SC-M	SC	Centres EDTracker
SC-D	PSB+SC	(not mapped)



Throttle SC (Slew Control Press Button)

(... Throttle LEVER continued)

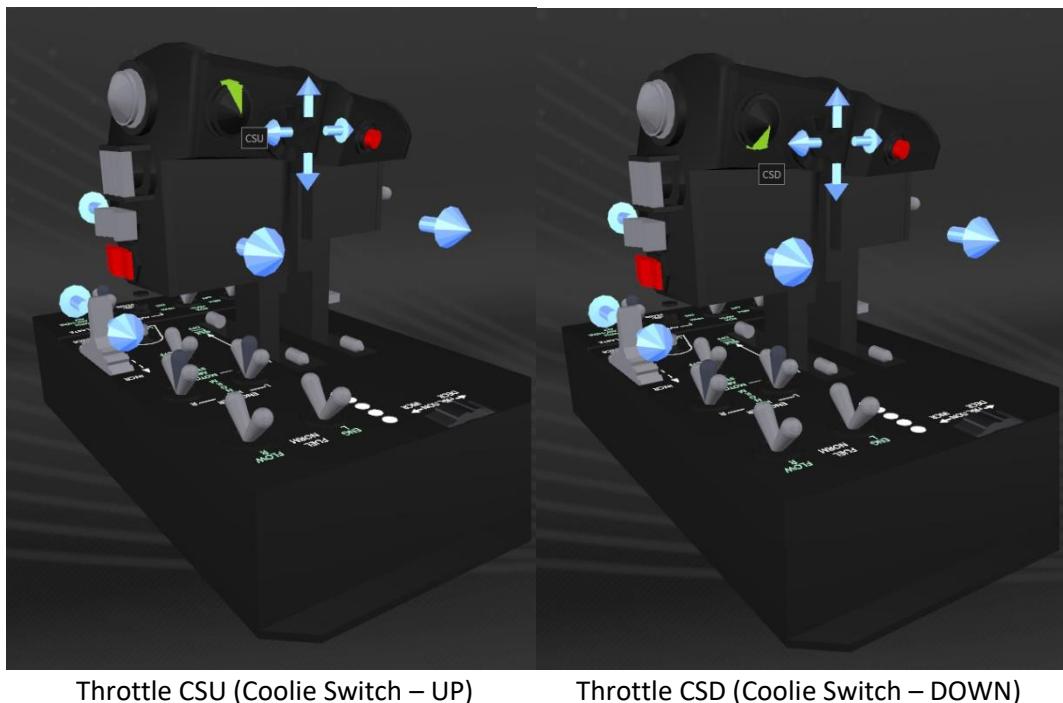
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
LTB-U	PSF+LTB	Toggles external ‘free camera view’ and turn off GUI
LTB-M	LTB	Toggle Head look ON/OFF
LTB-D	PSB+LTB	(not mapped)



Throttle LTB (Left Throttle Button)

(... Throttle LEVER continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
CSU	CSU	Engage/Cancel/Disengage Super cruise
CSD	CSD	Engage/Cancel System Jump (Hyperspace)



The script can detect a wide range of game states via status.json including if we are in Super cruise versus Hyper jump, when we are Mass Locked, when the FSD is cooling down or charging up.

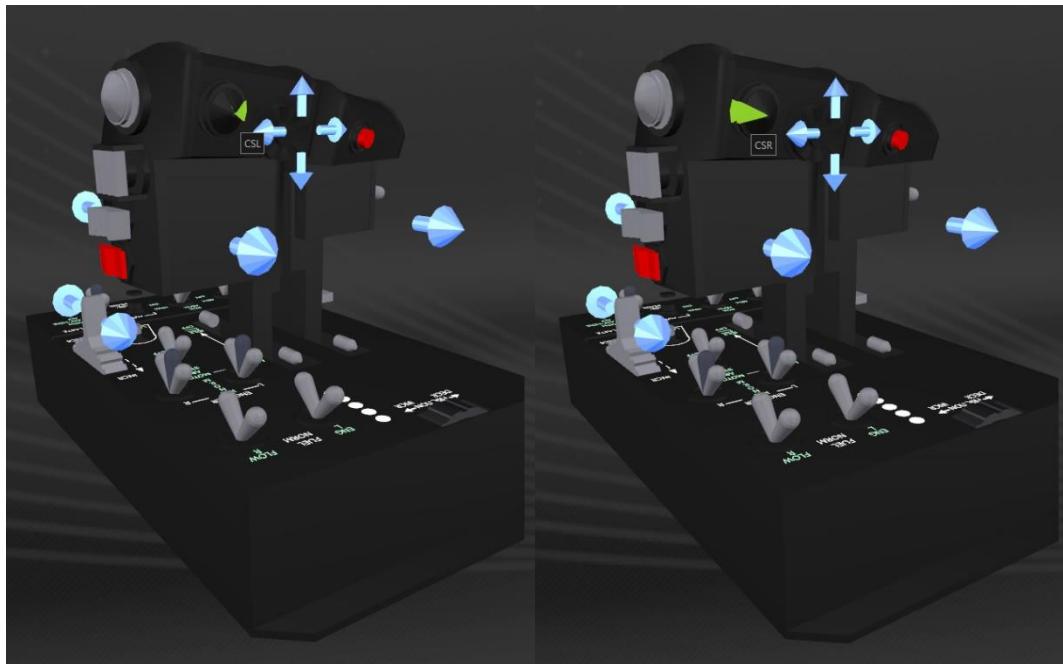
The script also detects when we drop out of Super cruise and if this was initiated or we have dropped out via interdiction, hyperdicted, or drop from orbital cruise etc.

When we Charge the FSD, the script will automatically retract the hard points, landing gear and cargo scoop if needed.

When interdicted, as soon as we drop from Super cruise, the script will automatically select Combat HUD mode, deploy hard points and endeavour to select the highest threat.

(... Throttle LEVER continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
CSL	CSL	Select Next System in route
CSR	CSR	Toggle Orbit Lines ON/OFF



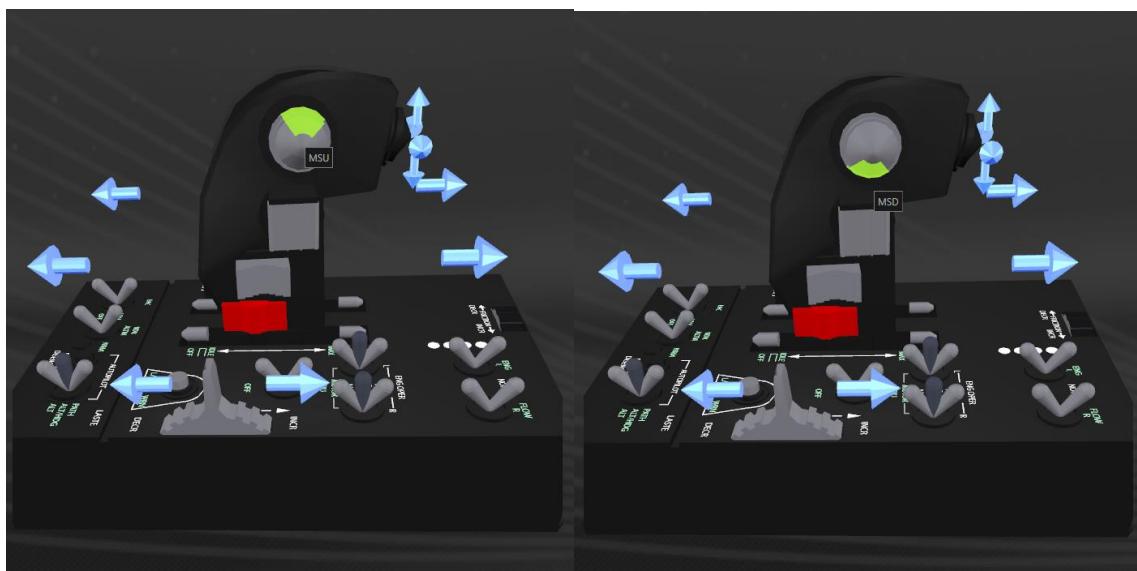
Throttle CSL (Coolie Switch – LEFT)

Throttle CSR (Coolie Switch – RIGHT)

(... Throttle LEVER continued)

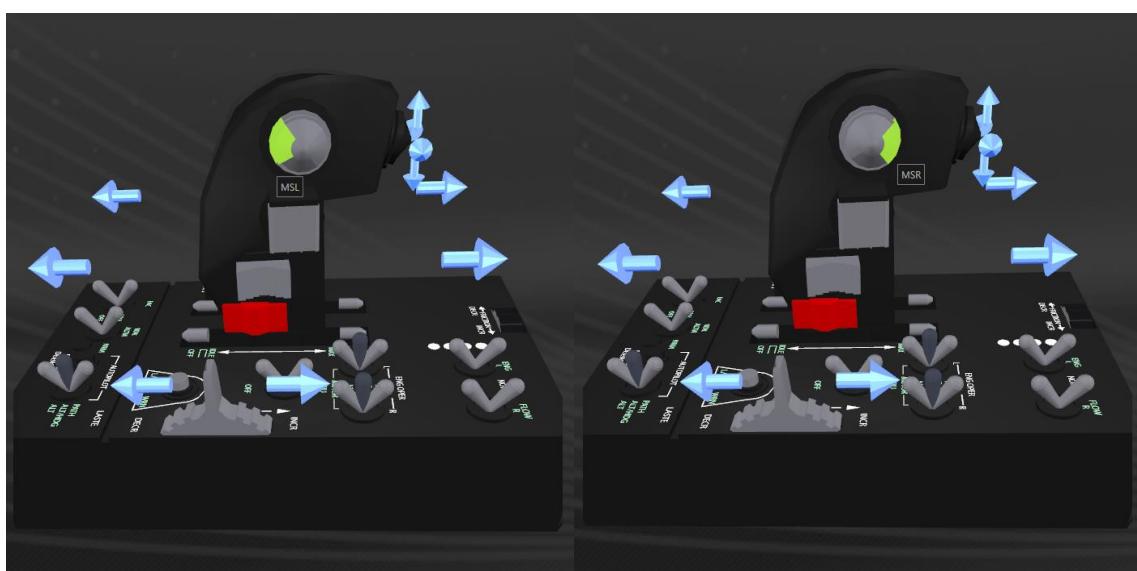
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
MSU	MSU	If TFRPRudder = 0, Alternate Roll Control (Counter clockwise)
MSD	MSD	If TFRPRudder = 0, Alternate Roll Control (Clockwise)
MSL	MSL	If TFRPRudder = 0, Alternate Yaw Control (Right)
MSR	MSR	If TFRPRudder = 0, Alternate Yaw Control (Left)

Note: 'TFRPRudder' setting is found in the ED_UserSettings.tmh file



Throttle MSU (Mike Switch – UP)

Throttle MSD (Mike Switch – DOWN)

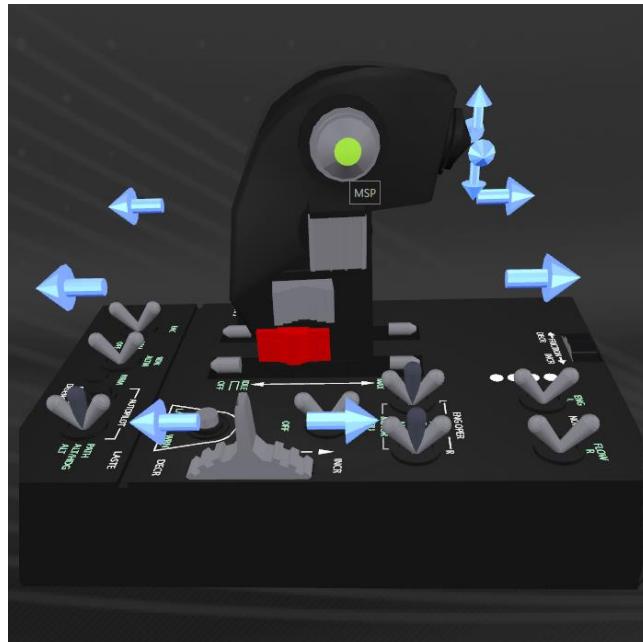


Throttle MSL (Mike Switch – LEFT)

Throttle MSR (Mike Switch – Right)

(... Throttle LEVER continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
MSP	MSP	(not mapped)



Throttle MSP (Mike Switch – PRESS)

The ‘Mike Switch’ is reserved for voice communications functions (future functionality) and as such the switches are not mapped to any actions by default.

If you do not have a set of rudder pedals, you can set ‘TFRPRudders = 0’ in the ED_UserSettings.tmh file. Doing so will map roll and yaw to MSU/MSD and MSL/MSR respectively.

(... Throttle LEVER continued)

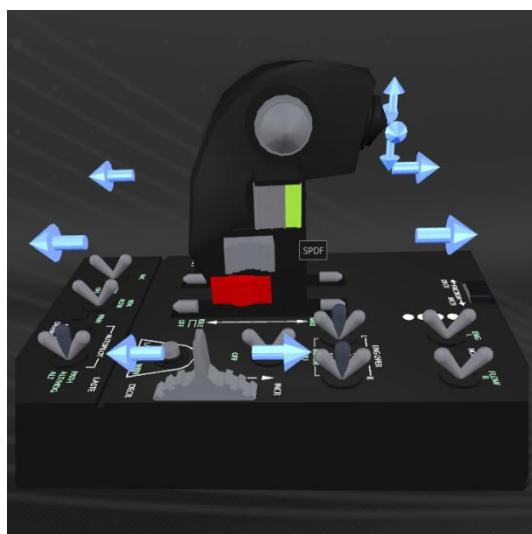
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
SPDF-U	PSF+SPDF	Reverse Thrust (no speed brake)
SPDF-M	SPDF	Reverse Thrust (+Speed brake if hard points are deployed)
SPDF-D	PSB+SPDF	Re-align switch to action if out of synch

The landing gear used as an auto speed brake by hitting ‘Reverse’ when in normal flight.

This will automatically deploy the landing gear, then retract 5 seconds later.

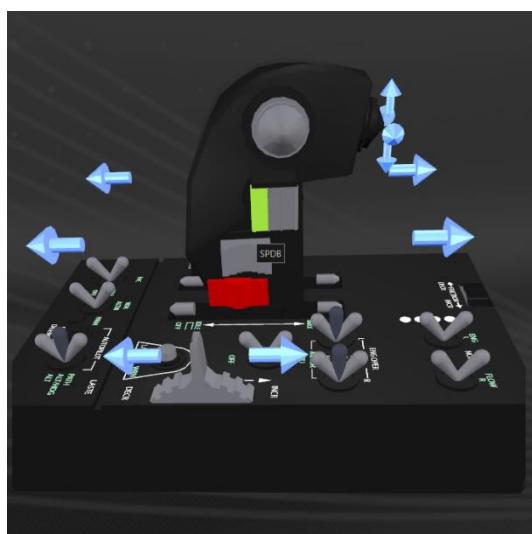
[Boost-bleeding](#) can be accomplished by hitting boost, then Reverse.

Reverse thrust turned off when switch returned to middle position.



Throttle SPDF (Speed Brake – FORWARD)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
SPDB	SPBD	Boost (momentary action switch)

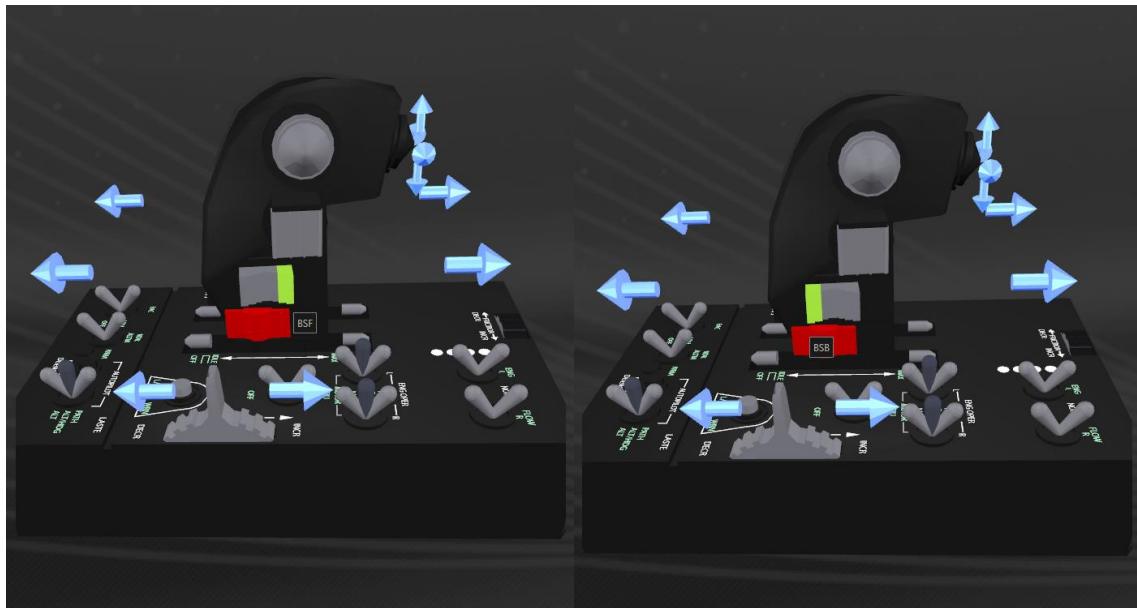


Throttle SPDB (Speed Brake – BACK)

(... Throttle LEVER continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
BSF-U	PSF+BSF	Macro: Toggles Wingman Beacon WING/OFF (Right Panel must be at home position or this won't work)
BSF-M	BSF	(not mapped)
BSF-D	PSB+BSF	Toggle Text-To-Speech feedback ENABLED/DISABLED

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
BSF-F	PSF+BSB	Toggle 'Report Crimes against me' ON/OFF (Right panel must be at home position or this won't work)
BSB-M	BSB	(not mapped)
BSB-D	PSB+BSB	(not mapped)



NOTE:

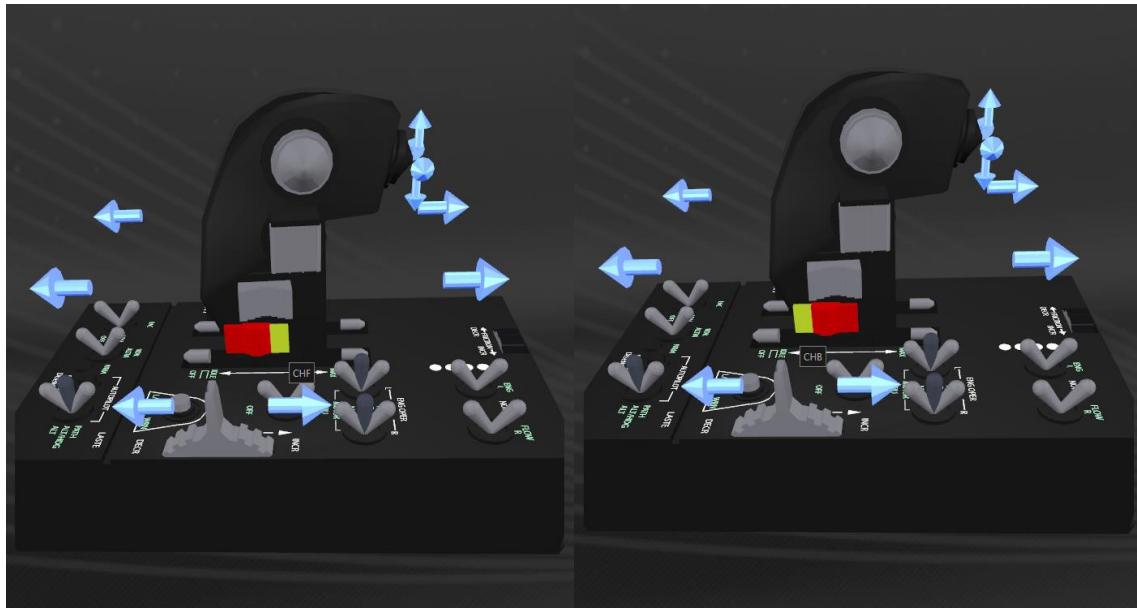
These macros require the SYSTEM Panel (Right Panel) to be at home position or undesired/random results will happen. Home position is the left most TAB in the Top Left position.

If you are used to manually accessing the panels frequently, consider not using these functions, or, get in the habit of returning to the home position manually each time.

This also applies to the 'Docking Request' function from the NAV (Target) Panel (Left Panel)

(... Throttle LEVER continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
CHF	CHF	Select next subsystem on targeted ship (momentary action switch)
CHB	CHB	Select previous subsystem on targeted ship (momentary action)



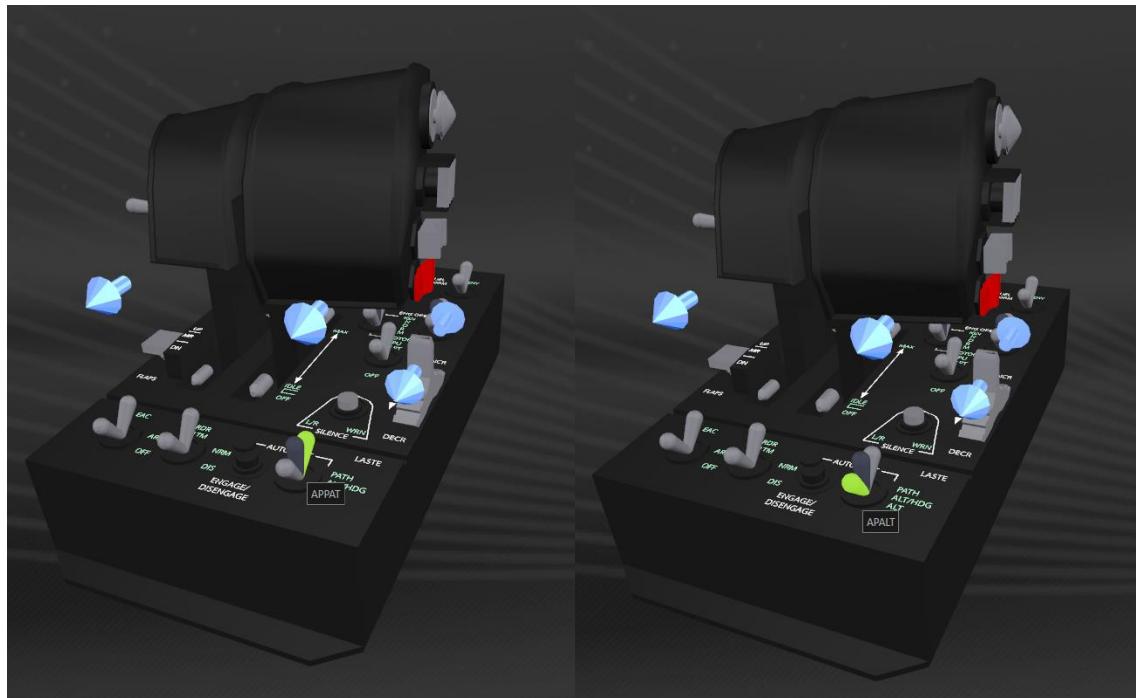
Throttle CHF (China Hat – FORWARD)

Throttle CHB (China Hat – BACK)

Throttle Control Base

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APPAT-O	APPAT	Set Joystick/Rudder curves to MILD
APPAT-I	S3+APPAT	Set Joystick/Rudder curves to MEDIUM
APAH	APAH	Set Joystick/Rudder curves to OFF
APALT-O	APALT	Set Joystick/Rudder curves to SLOW
APALT-I	S3+APALT	Set Joystick/Rudder curves to SLOWEST

Refer to Axis Curves and Curve Control

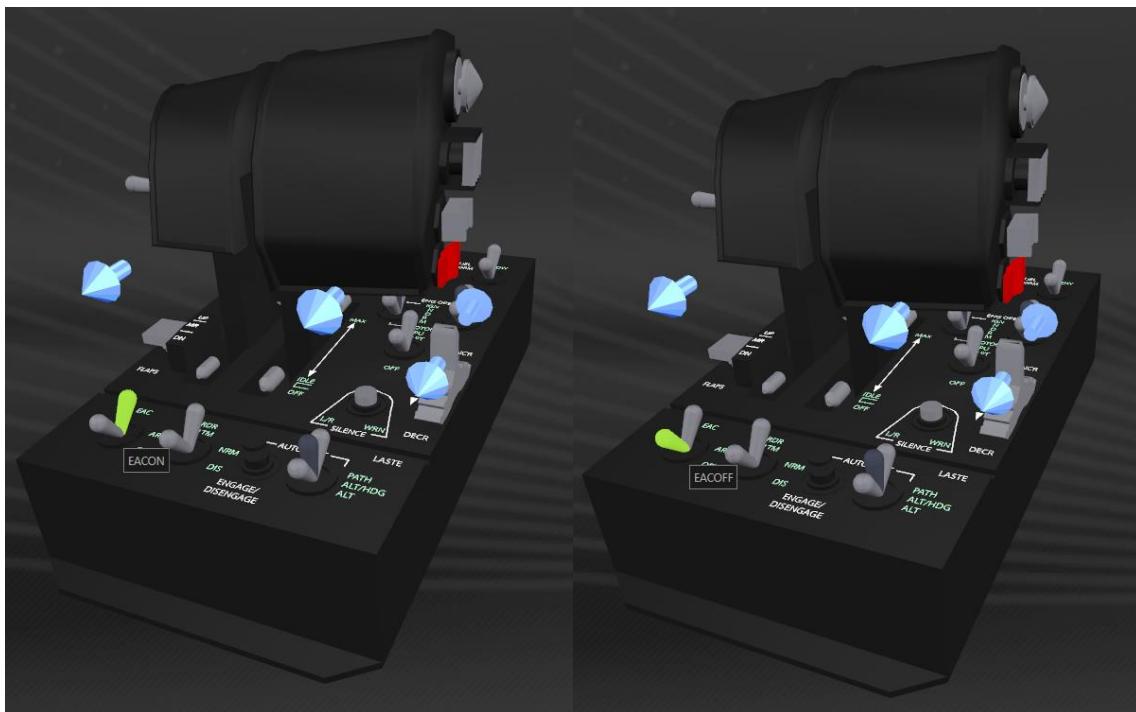


Throttle APPAT (Auto Pilot - Path)

Throttle APALT (Auto Pilot - ALT)

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EACON-I	S3+EACON	Toggle Silent Running – STROBE
EACON-O	EACON	Toggle Silent Running – ON
EACOFF	EACOFF	Toggle Silent Running – OFF



Throttle EACON (Enh Attitude Control – ON)

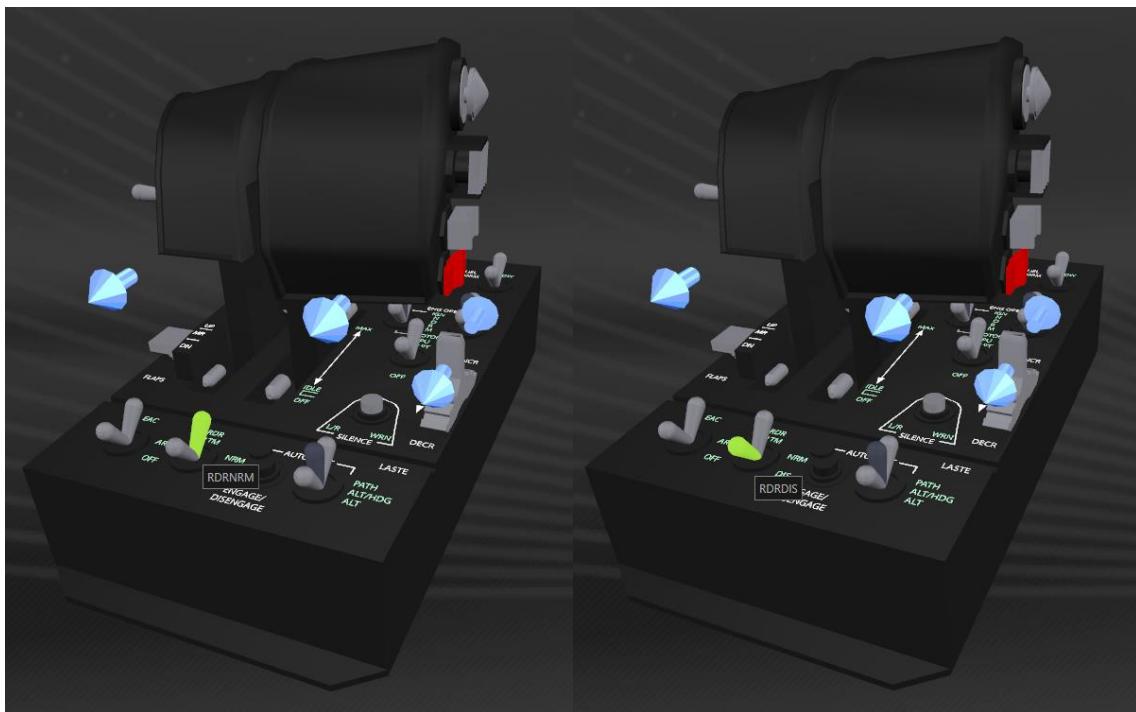
Throttle EACOFF (Enh Attitude Control – OFF)

Silent Running: Status LED 3

STROBE mode toggles Silent Running on/off every 2 seconds. This breaks target lock of attacking ships.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
RDRNRM-IU	S3+PSF+RDRNRM	Jettison Cargo
RDRNRM-OU	PSF+RDRNRM	(not mapped)
RDRNRM-IM	S3+RDRNRM	(not mapped)
RDRNRM-OM	RDRNRM	Deploy Cargo Scoop
RDRNRM-ID	S3+PSB+RDRNRM	(not mapped)
RDRNRM-OD	PSB+RDRNRM	(not mapped)
RDRDIS	RDRDIS	Retract Cargo Scoop



Throttle RDRNRM (Radar Altimeter – NRM)

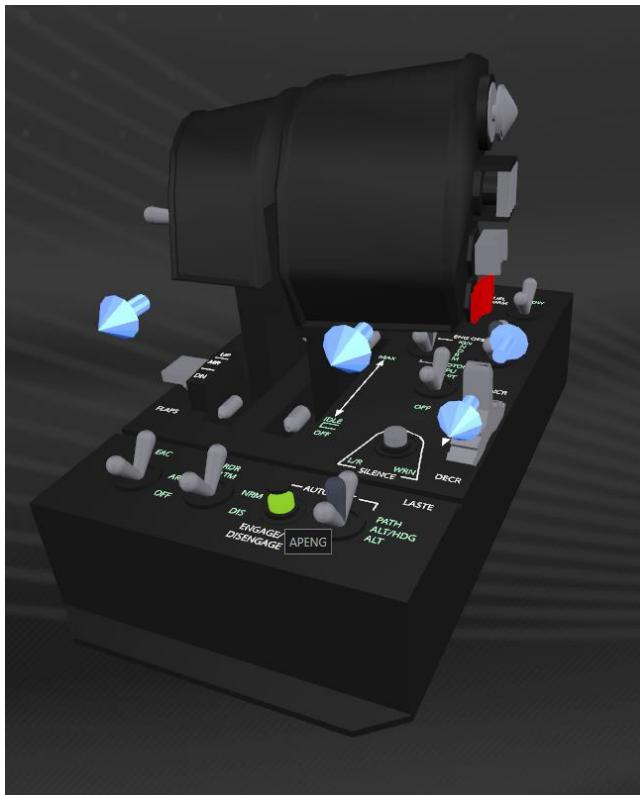
Throttle RDRDIS (Radar Altimeter – DIS)

Cargo Scoop: Status LED 4

Jettison Cargo: Jettisons all cargo in cargo hold.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APENG-I	S3+APENG	Enter Hanger and go to Hanger Services
APENG-O	APENG	Request docking permission (NAV/Target Panel must be in home position or this will not work)



Throttle APENG (Autopilot Engage Button)

Request docking macro fires a sequence of keystrokes to request docking then returns the selector back to the 'home position' in the NAV panel.

When you move within the panels manually, there is no tracking mechanism and my code will not know if we are starting from the 'home' position.

If the NAV panel is not at the 'home position' when you hit this switch, unpredictable results will happen...USE WITH CAUTION!

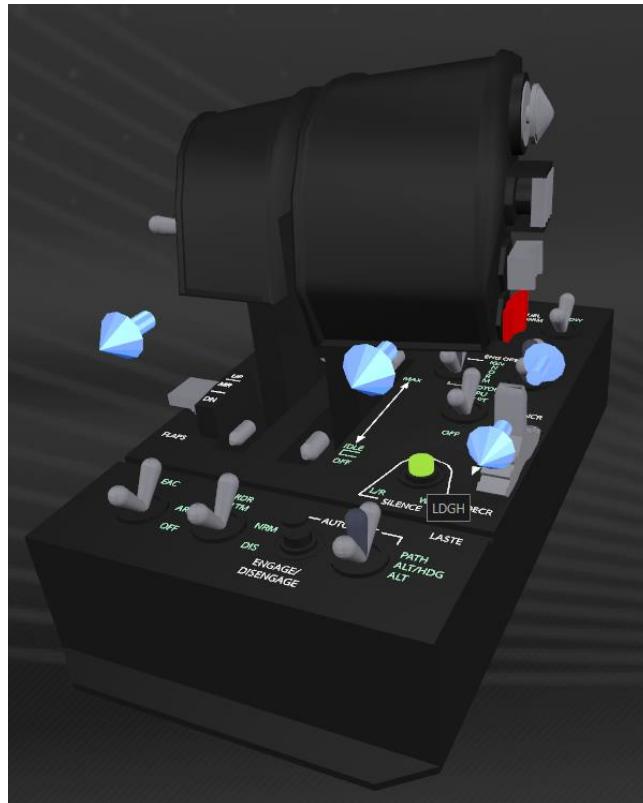
The docking request macro will also set 4x PIPs to SYS and 2x PIPs to ENG. If 'AutoGearDOWN' is set in ED_UserSettings the landing gear will deploy automatically.

The script also detects when you touch down and will balance PIPs one second after you touch down.

If 'AutoGearUP' is set in ED_UserSettings, the landing gear will automatically retract when you lift off.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
LDGH-I	S3+LDGH	Clear all three Chat box windows
LDGH-O	LDGH	Clear currently selected Chat box



Throttle LDGH (Landing Gear Horn Silence Button)

The 'clear chat box' macros send the following key sequence to the selected chat box;

enter, /, c, l, e, a, r, enter

(... Throttle BASE continued)

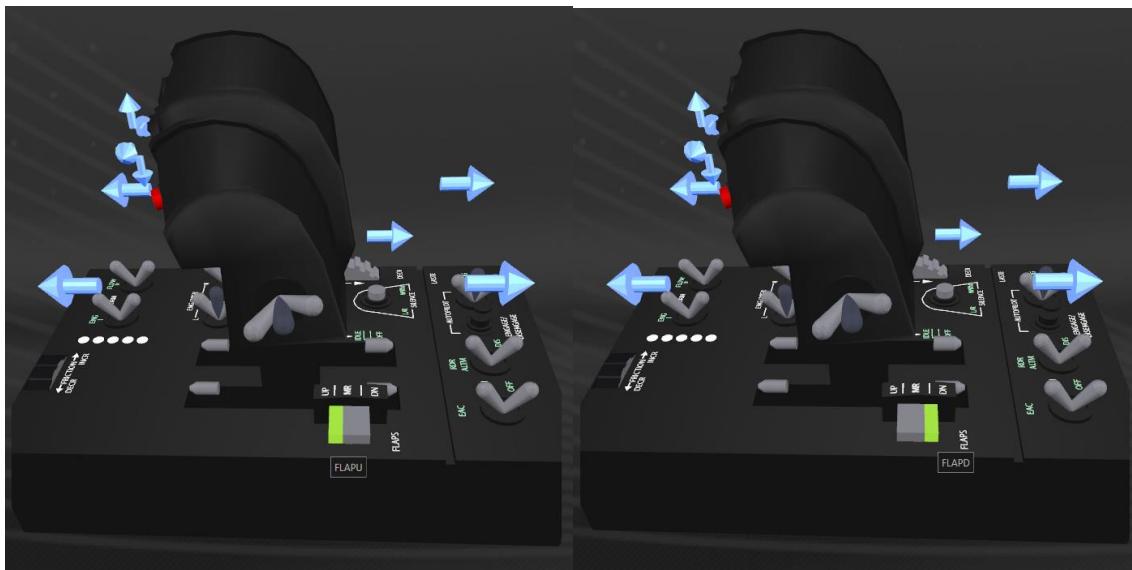
<u>Switch</u>	<u>Usage</u>	<u>Action</u>
FLAPU-U	PSF+FLAPU	Increase Text-To-Speech Volume
FLAPU-M	FLAPU	Go to Galaxy Map
FLAPU-D	PSB+FLAPU	Increase Throttle LED Brightness (0 → 5 → 0)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
FLAPD-U	PSF+FLAPD	Decrease Text-To-Speech Volume
FLAPD-M	FLAPD	Go to System Map
FLAPD-D	PSB+FLAPD	Turn Throttle LEDs OFF
FLAPDR-D	PSB+FLAPDR	Turn Throttle LEDs back ON

Increase and decrease Text-To-Speech Volume changes volume of the Voice feedback in increments of 5%. Example: To lower the volume by 10%, use PSF+FLAPD twice. (Lowest volume = 5%)

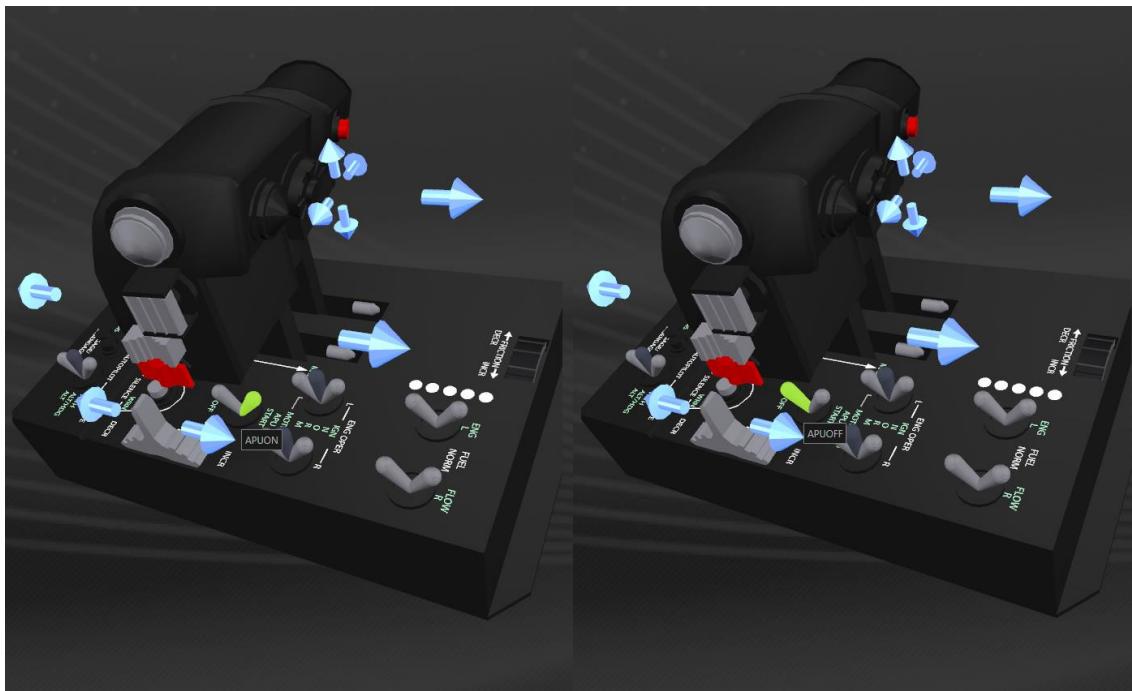
Turn LEDs OFF by PSB+FLAPD. If you move the PSF modifier switch back to its default PSM position then move the Flaps switch back to M, the LEDs stay off.

To turn them back on, simply toggle PSB+FLAPD and back to middle.



(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
APUON	APUON	Deploy SRV ROLE Panel must be in home position or this will not work
APUOFF	APUOFF	Recover SRV ROLE Panel must be in home position or this will not work



Throttle APUON (Aux Power Unit – START)

Throttle APUOFF (Aux Power Unit – OFF)

DEPLOY SRV:

The script cannot check the presence of an SRV.

If you use this function with no SRV, unpredictable results will happen.

The ROLE Panel (lower panel) must be in the Home position for this macro to work.

The script checks that you are landed, however, make sure you have an SRV!

RECOVER SRV:

The script cannot currently detect when you are directly under the ship ('boarding light' on).

It will check when the turret is stowed (which means you are close).

If you use this function when not directly under the ship, unpredictable results will happen.

The ROLE Panel (lower panel) must be in the Home position for this macro to work.

Make sure the 'Boarding Light' is lit before flicking the APUOFF switch.

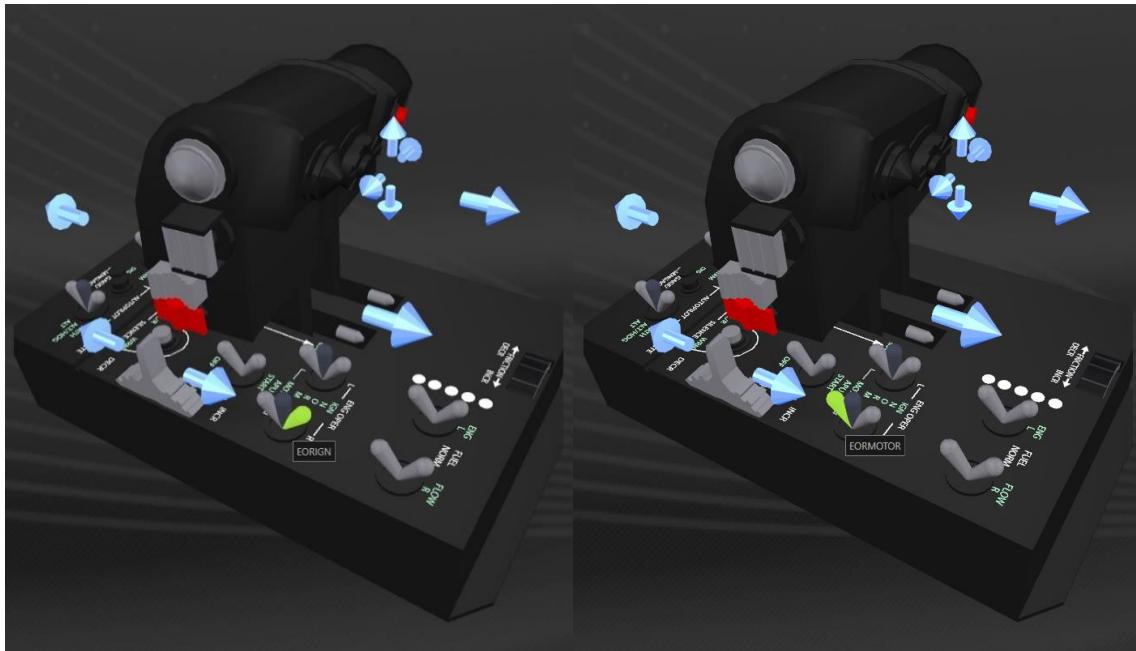
WARNING:

I've seen some unpredictable behaviour in the RECOVER macro at times and am yet to make this macro 100% reliable.

This script macro positions the selector but does not send 'enter' in case it is not at the correct menu option.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EORIGN	EORIGN	Dismiss/Recall Ship (momentary action)
EORMOTOR	EORMOTOR	Toggle Ship GUI OFF/ON



Throttle EORIGN (Engine Operate – IGNITION) Throttle EORMOTOR (Engine Operate – MOTOR)

Dismiss/Recall Ship:

This function is only useful if you are not planning to stray too far away from your landed ship, or if you are in the habit of dismissing your ship every time you deploy the SRV.

The Flags key value in status.json does not have a bit representing if the Ship is landed nearby or if it was dismissed. I have implemented my own status variable for this, however if you manually dismiss the ship via keyboard or mouse interaction, or if the ship flies off after you wander far enough away, this variable will not be able to be tracked.

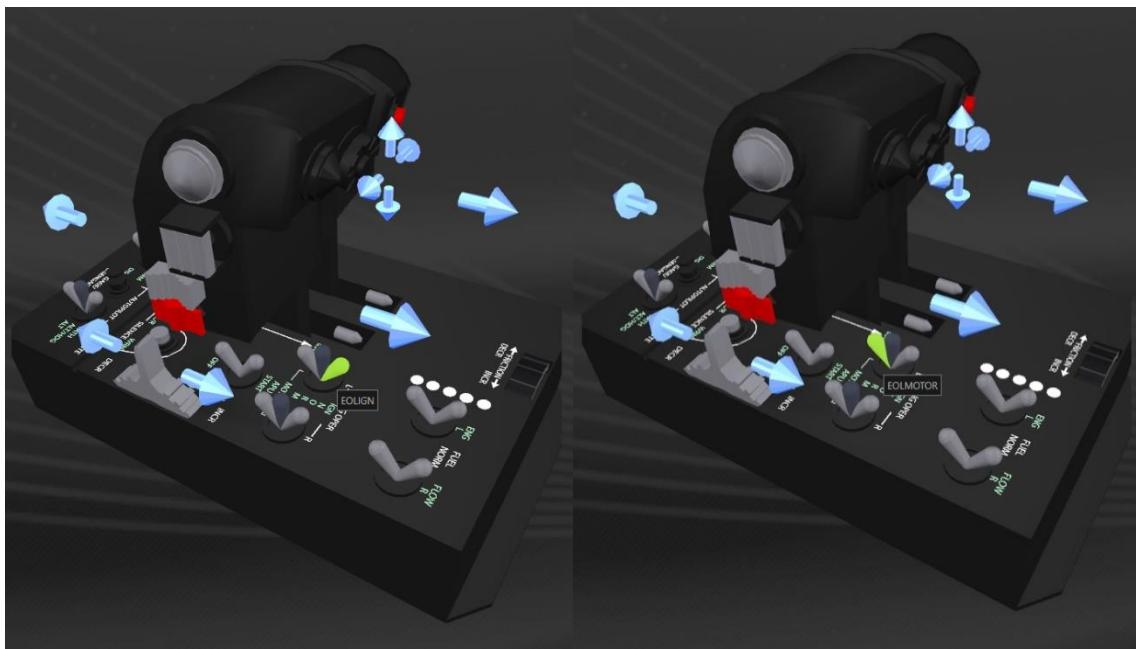
This results in the voice feedback potentially being wrong.

I have turned off voice feedback for this function for now, however the code still exists.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EOLIGN-U	PSF+EOLIGN	Game Mode Switch to OPEN
EOLIGN-M	EOLIGN	Game Mode Switch to PRIVATE GROUP
EOLIGN-D	PSB+EOLIGN	Game Mode Switch to SOLO

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EOLMOTOR	EOLMOTOR	Toggle Engine and Weapon Colours



Throttle EOLIGN (Engine Operate – IGNITION) Throttle EOLMOTOR (Engine Operate – MOTOR)

Game Mode Switch:

This macro exits the game to the main menu and logs back in to which ever mode you selected via the PSF/PSM/PSB switch.

The script checks for and sets various ship systems that are reset as part of a Menu log or game restart.

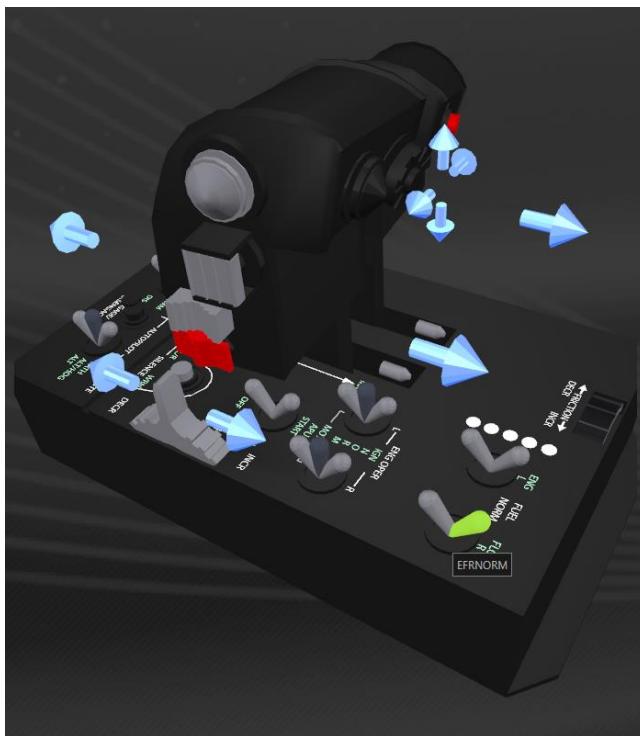
These include the Cargo Scoop, Reverse Thrust and Silent Running.

Last time I checked, FA-OFF and the condition of the lights, are not reset.

NOTE: This Mode Switching macro will not fire if you are in danger (ie Combat) and so cannot be used to Combat Log (clog) on your opponent.

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EFRNORM-U	PSF+EFRNORM	Reset 'GameLoaded' variable (see note below)
EFRNORM-M	EFRNORM	Reset Status LEDs (see note below)
EFRNORM-D	PSB+EFRNORM	Print Status Page banner to console



Throttle EFRNORM (Fuel Flow Right – NORMAL)

GameLoaded Variable:

Script Variable that checks when/if the game is loaded.

This is a legacy function used when Game Mode switching sometimes did not work as expected.

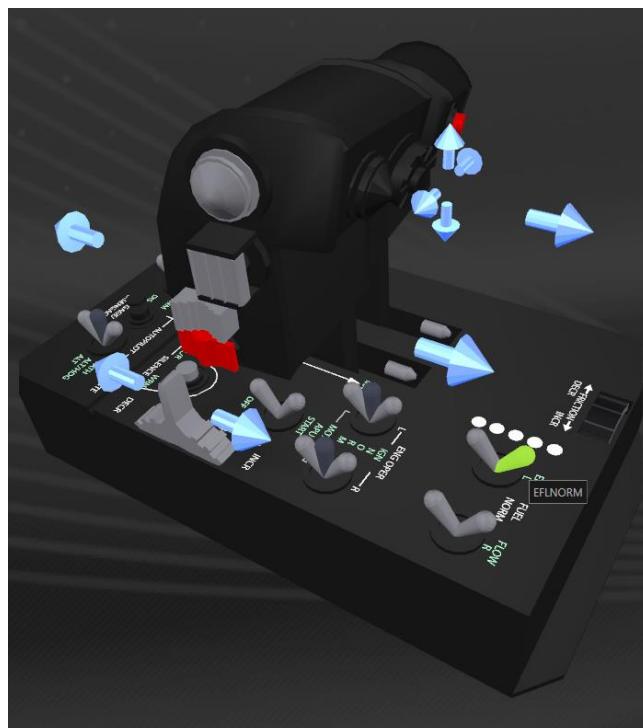
Reset Status LEDs:

If the Status LEDs are toggled in quick succession, they can occasionally go out of synch. Using EFRNORM will force the script to align the LEDs to the values of status.json 'Flags' bits.

LED	Status	LED on	flag bit variable
LED1	Flight Assist	OFF	fFAOff
LED2	Ship/SRV Lights	ON	fLights
LED3	Silent Running	ON	fSilentRunning
LED4	Cargo Scoop	DEPLOYED	fCargoScoop
LED5	Landing Gear	DEPLOYED	fLandingGear

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
EFLNORM-U	PSF+EFLNORM	Debug Tool. Turn 'ShowFlags = ON'
EFLNORM-M	EFLNORM	Debug Tool. Turn 'Debug = ON'
EFLNORM-D	PSB+EFLNORM	Debug Tool. Turn 'Debug' and 'ShowFlags' = ON
EFLOVER	EFLOVER	Debug Tool. Turn 'Debug' and 'ShowFlags' = OFF



Throttle EFLNORM (Fuel Flow Left – NORMAL)

TARGET GUI

LAUNCHING AND RUNNING PROFILE

STOP PROFILE

Silent Running: ACTIVATED !!
Warp Drive: MASSLOCKED <<
!!! Function: Warp Drive not available at this time, we are currently masslocked !!!
Ship Lights: ON >>
!!! Function: Landing Gear not available at this time, we are landed !!!
Warp Drive: MASSLOCKED <<
!!! Function: Warp Drive not available at this time, we are currently masslocked !!!

= Docking request sent =
= Power to shields =

Ship System States @ 15:43 JSON-Timestamp: 2020-01-28T06:37:46Z Flags: 18939917

MAIN SHIP:	DOCKED	Shields:	UP	Mass Locked:	YES
Altitude:	3456	Heading:	012	Lat/Long:	35.894, -36.897
HUD Mode:	COMBAT	Ship Lights:	OFF	Night Vision:	OFF
Cargo Scoop:	RETRACTED	Landing Gear:	DEPLOYED	Hardpoints:	RETRACTED
FSS Mode:	OFF	Supercruise:	DISENGAGED	Silent Running:	OFF
Flight Assist:	OFF	Reverse Thrust:	FORWARD	Speed Brake:	RETRACTED

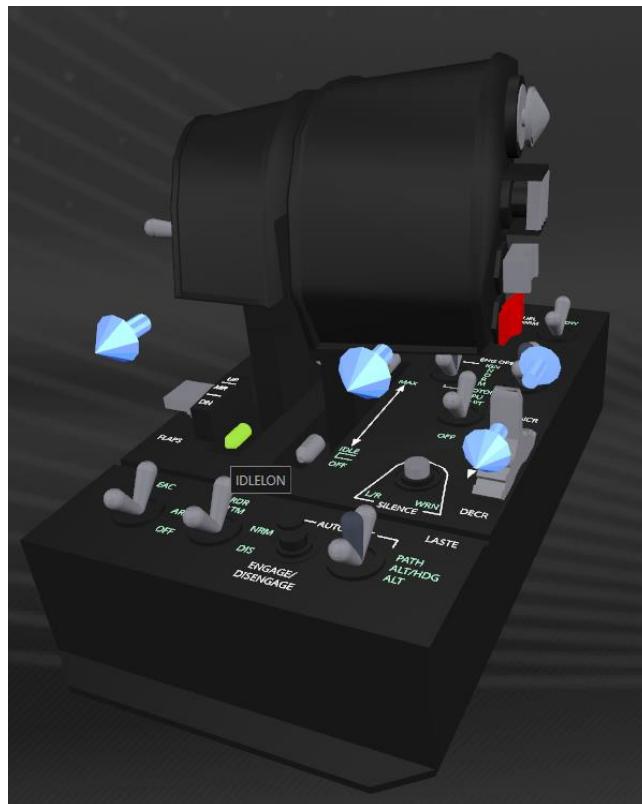
TEST TOOLS

DEVICE ANALYZER EVENT TESTER JOYSTICK CONTROL PANEL

Example of Status Page

(... Throttle BASE continued)

<u>Switch</u>	<u>Usage</u>	<u>Action</u>
IDLELON	IDLELON	Ship: Enter Hanger and Station Services SRV: Toggle hand brake on
IDLEOFF	IDLEOFF	Ship: Exit Hanger and Lift off (retract landing gear) SRV: Toggle hand brake off



Throttle IDLELON (Engine Idle – LEFT)

Engine Idle ON LEFT/RIGHT:

To engage Engine Idle Switch, from the Throttle all the way back position, LIFT and move back over the notch.

Engine Idle OFF LEFT/RIGHT:

From the Engine Idle ON position (lifted + all the way back), move the Throttle lever forward until it drops back off the notch.

TFRP Rudder Controller

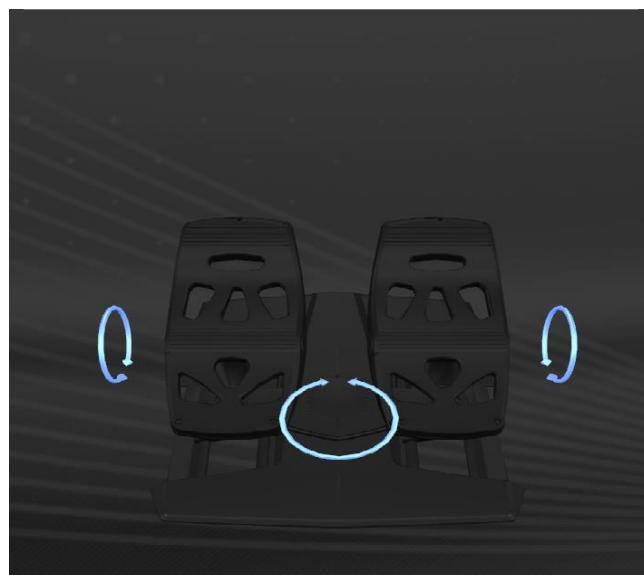
The rudder pedals provide yaw for the ship and steering for the SRV.

Curves may be applied to the rudder via the autopilot switch and can be customised within the ED_UserSettings.tmh file.

Curves are turned off by flicking the Autopilot switch to the 'ALT/HDG' position (APAH).

These curves are linked to the Joystick curve profile.

Refer to the 'Axis Curves and Curve Control' section.



TFRP Rudder Pedals

Note:

The Toe Brakes are not currently mapped in this script.

Keyboard Key Binds

Notable keys (generally no script function mapped):

<u>Key</u>	<u>Label</u>	<u>Action</u>
RSHIFT+W	RollLeft	Alternate Roll Control – Left (mapped if no rudders)
RSHIFT+S	RollRight	Alternate Roll Control – Right (mapped if no rudders)
RSHIFT+Q	YawLeft	Alternate Yaw Control – Left (mapped if no rudders)
RSHIFT+E	YawRight	Alternate Yaw Control – Right (mapped if no rudders)
NP-3	SetSpeed75	Sets Throttle to 75% (Blue Zone)
X	SetSpeed0	Set Throttle to 0 (or SRV Handbrake)
NP-Enter	SwapRollYaw	Swaps Roll axis and Yaw Axis (Mapped to S1-D)
LALT+B	FighterDock	Send Dock order to fighter
LALT+D	FighterDefend	Send Defend order to fighter
LALT+A	FighterAggr	Send Aggressive order to fighter
LALT+T	FighterAttack	Send Attack order to fighter (focus on my target)
LALT+H	FighterNoFire	Send Hold fire order to fighter
LALT+S	FighterHold	Send hold position order to fighter
LALT+F	FighterFollow	Send follow me order to fighter
LALT+O	FighterOpen	Send open orders to fighter
LALT+W	WeaponColour	Swaps Weapon Colour On/Off
LALT+E	EngineColour	Swaps Weapon Colour On/Off
RALT+F11	GalNetPlay	Toggle Play/Pause GalNet Audio
RALT+F10	GalNetPrev	Select previous GalNet item to play
RALT+F12	GalNetNext	Select next GalNet item to play
RALT+F9	GalNetClear	Clear the GalNet Audio Queue
LCTRL+1-9	CameraPreset1-9	Cycle Camera pre-sets

Note:

Multi crew key binds are not currently used in this script nor mapped in the bind files

Fighter commands are not currently used in the script, however these are present in the bind files

(... Keyboard Keybinds continued)

Full Keybind Listing Refer to following files;

<u>File</u>	<u>Location</u>
ED_Defines.ttm	Script folder
KeyBinds.xls	Maps folder
Keyboard Usage.png	Maps folder

...or the below pages;

Joystick Map – BASIC and FULL

Throttle Map – BASIC and FULL

Key Bind Sheet

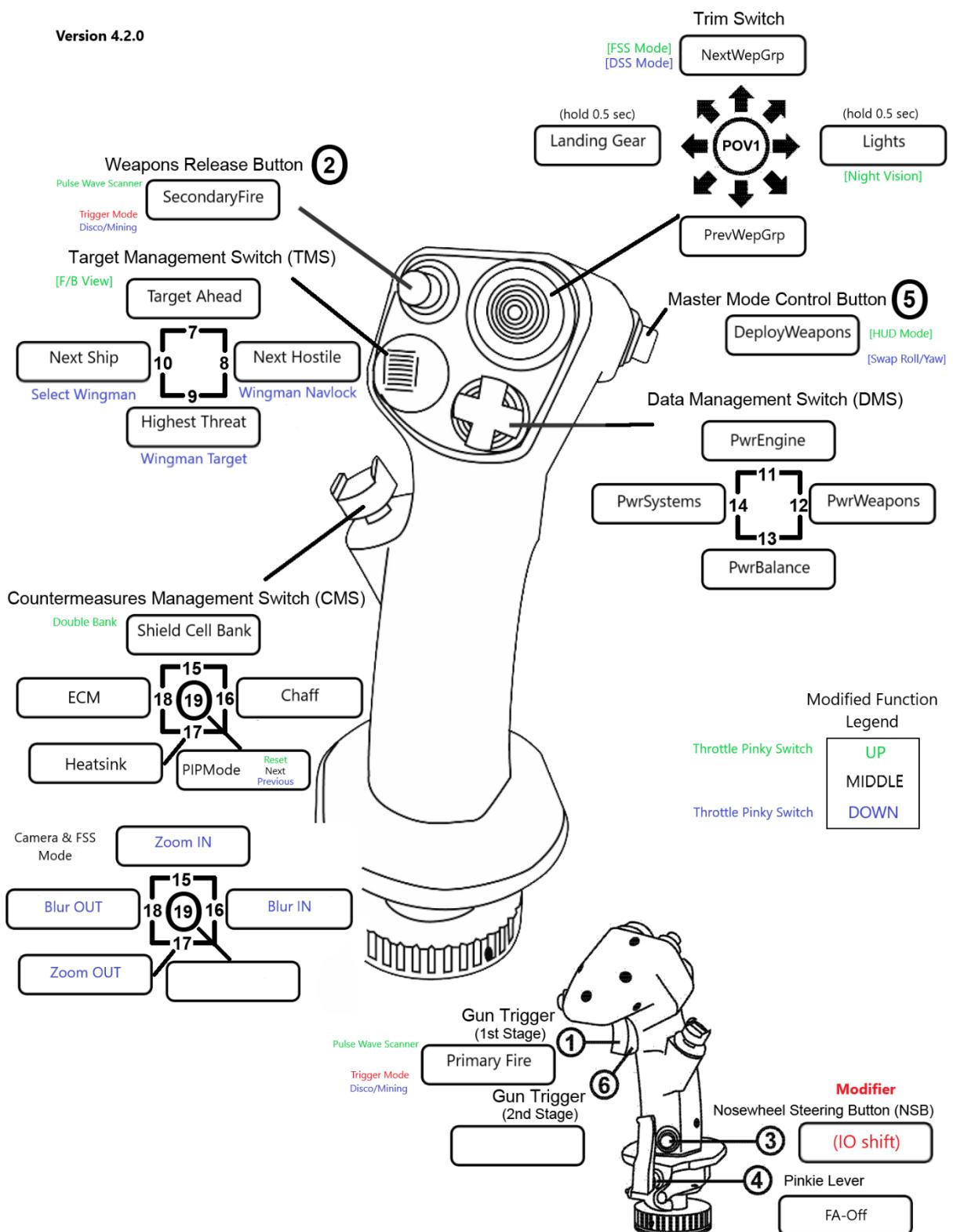
ED_UserSettings File

<u>Variable</u>	<u>Default</u>	<u>Use</u>
MapKeyProfile	FULL	Use all macros and functions
FlashLEDs	1	Enables flashing base LEDs at startup and status LEDs when status changes
LEDLevel	1	Sets the LED brightness level on the Throttle base (0 – 5)
StatusFile	<string>	Pathname for Elite Dangerous 'status.json' file
StatusReadRate	500	Loop timer (in msec) that the script will read status.json
MyStatusFile	<string>	Pathname for extra state flags to be saved (Refer to chapter on "Special Functions" later in this guide)
EnableVoice	1	Enables the TextToSpeech engine (voice feedback)
VoicePath	<string>	Pathname to 'voice.exe'
VoiceVolume	75	Default volume for TextToSpeech Voice
DefaultVoice	<alias>	Set to your preferred (installed) MS TTS Voice
VoiceCMD	<string>	Command line options for voice.exe
TrainingEnabled	1	Enables 'training mode'. When game is not running, pressing a button or switch will announce what it does via voice feedback and console printout
AnnounceTraining	1	Announces if Training Mode is available/active
EnableSoundFX	1	Enables the SoundFX Engine (.wav file player)
SoundPath	<string>	Pathname to 'sounder.exe'
WAVPath	<string>	Pathname to wav files for SoundFX
SoundCMD	<string>	Command line options for sounder.exe
DefaultPIPMode	3	Select your choice of 6 PIPModes as the default
PulseTime	33	Key press (hold down) time (in msec)
DelayTime	75	Delay between releasing one key and press the next (msec)
LongPress	500	Used for 'TEMPO' delay (msec). Hold key for 'LongPress' to trigger second action
DiscoDelay	6100	Trigger mode for Discovery Scanner. Holds trigger for 'DiscoDelay' milliseconds then releases (6.1 seconds)

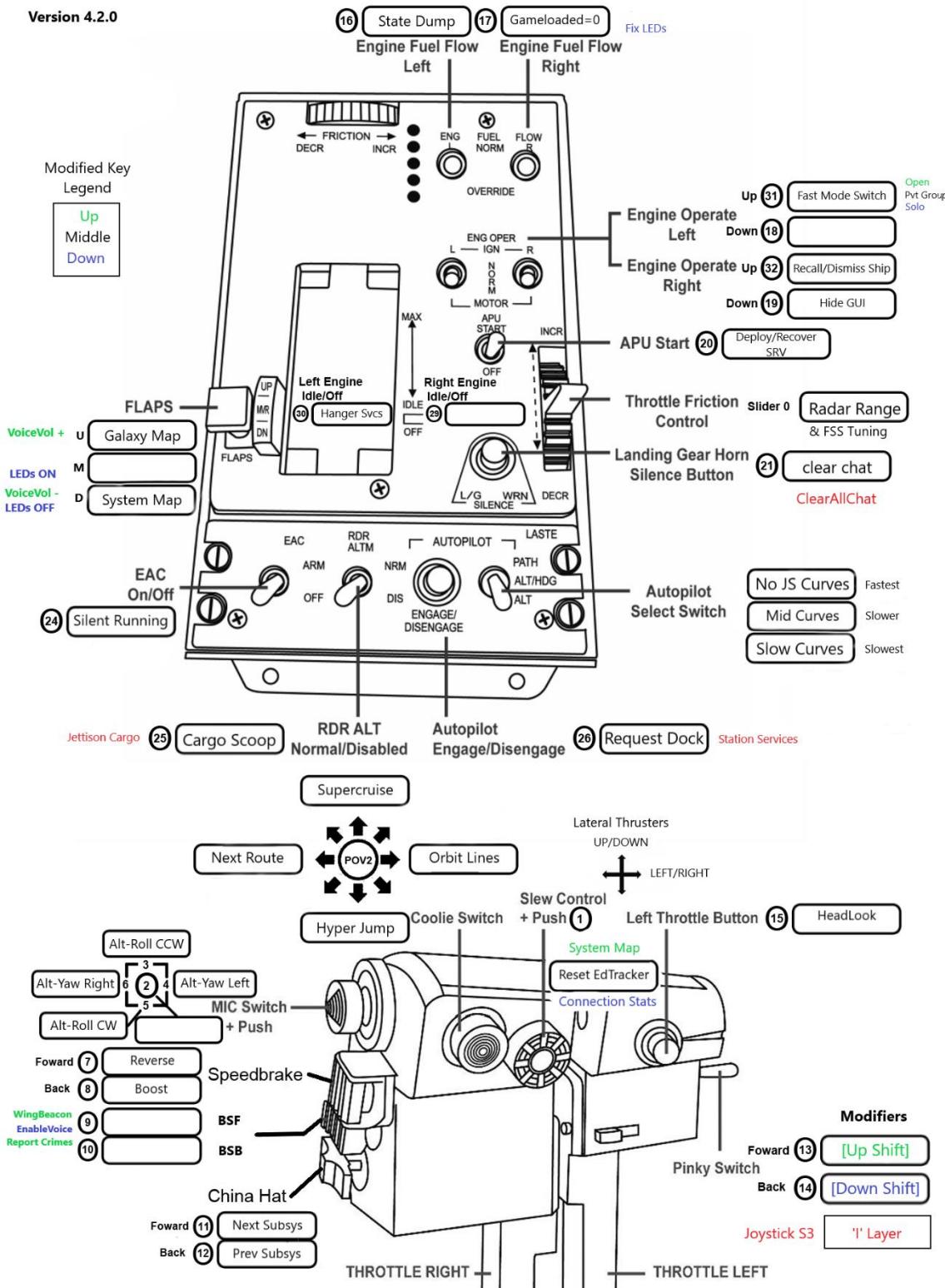
(... Ed_UserSettings File continued)

AutoHanger	0	When enabled will enter hanger and select station services automatically when you dock
AutoHUD	0	Enable to automatically engage Analysis Mode when going into super cruise
AutoGearUP	0	Raises Landing Gear automatically when you lift off
AutoGearDOWN	0	Lowers the Landing gear when you request docking perm
AutoHeatsink	1	Fire a heatsink when temperature hits 100%
AutoStrobe	1	Enable to strobe silent running on and off every 2 second when shields drop and 'in-danger'. Handy to break target lock on pursuing ships
CommsModule	0	Future feature – currently in test
Joystick_*	*	Joystick dead zone and zoom settings (for curves)
Rudder_*	*	Rudder dead zone and zoom settings (for curves)
Slew_*	*	Slew control dead zone and zoom (for curves)
FAOFFCurves	SLOW	Set your desired curve profile when you go FA-OFF
CurveInfo	NOPRINT	Will print curve profile settings to console if = PRINT
EnableTFRPRudder	1	Set this to '0' if you do not have these rudder pedals Alternate Roll and Yaw will then be available via Mike Switch
SwapRoll4Yaw	0	Set to 1 to swap the Roll and Yaw axes on Joystick/Rudder
Joystick_Curve[]	*	Joystick Curve array settings (0 - 4)
Rudder_Curve[]	*	Rudder Curve array settings (0 - 4)
Slew_Curve[]	*	Slew Curve array settings (0 or 2)

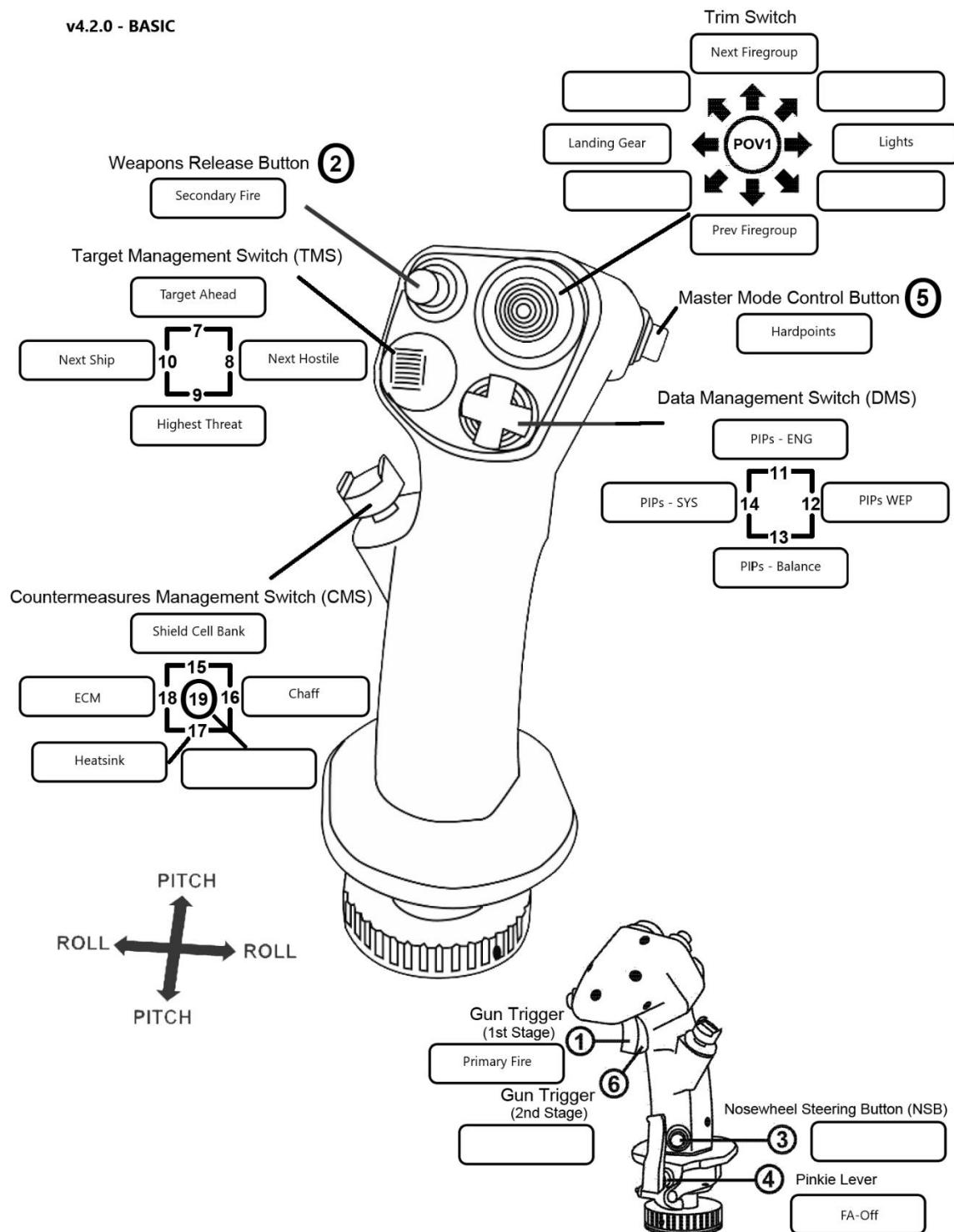
Joystick Map - FULL



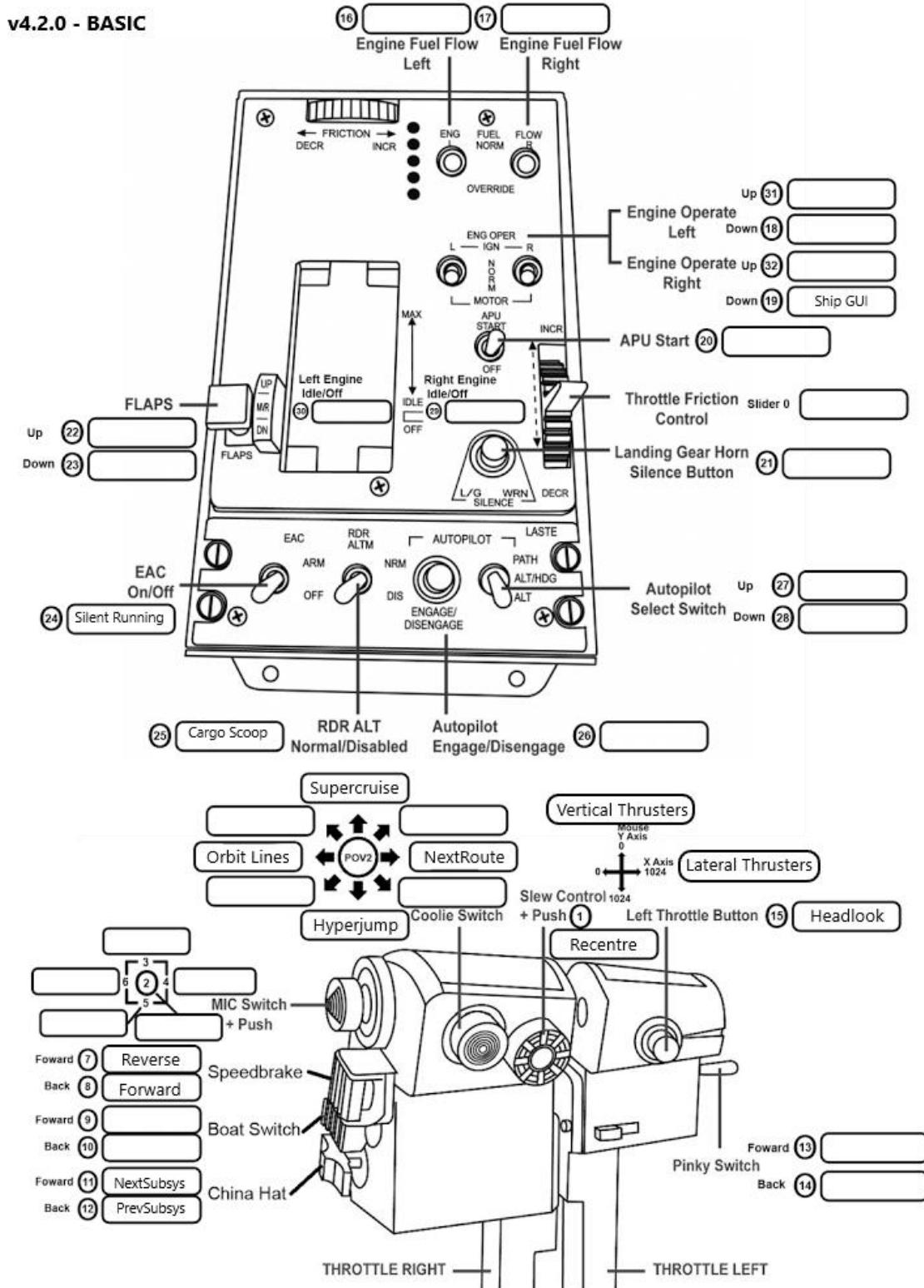
Throttle Map - FULL



Joystick Map – BASIC



Throttle Map – BASIC

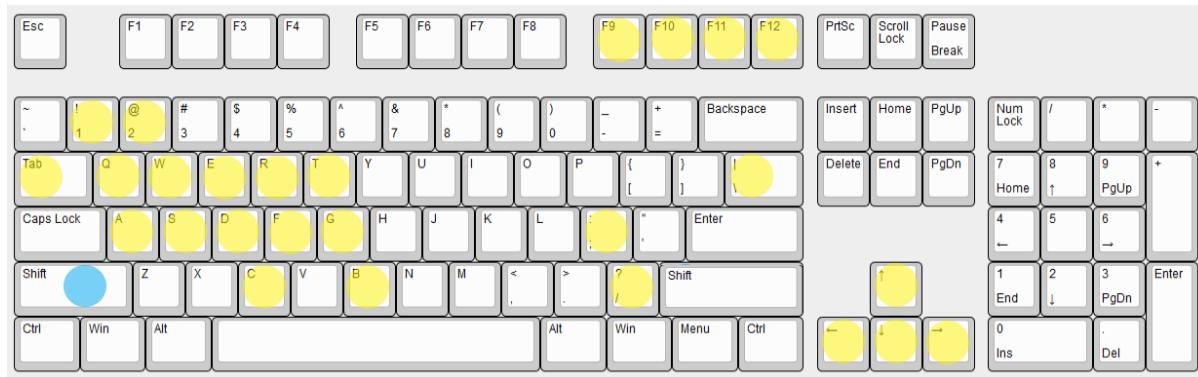


Keyboard Utilisation

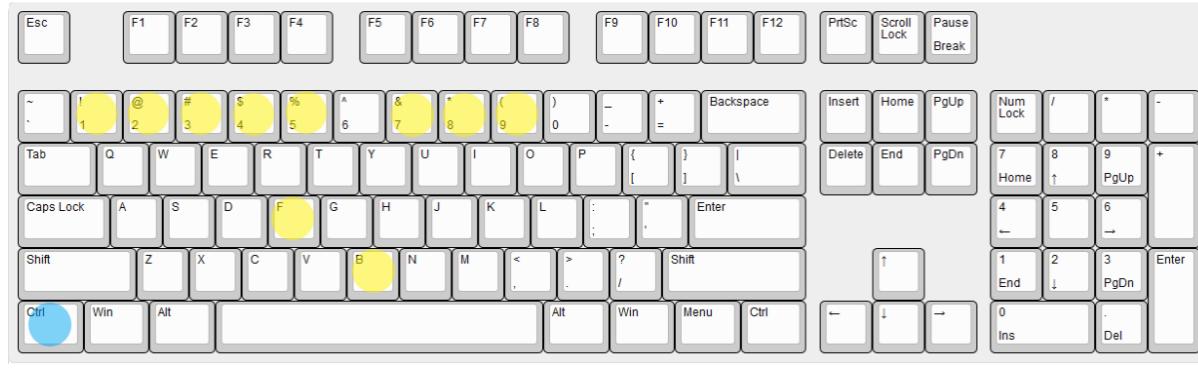
Normal (straight) Keypress



Modified Keypress



Modified Keypress



Modified Keypress



Key Bind Sheet

HOTAS Switch and Button Reference - FULL

HOTAS - Quick reference - v4.2.0																
Joystick				Throttle (stick)				Throttle (base)								
Switch	Mode	Function	Use for	Switch	Mode	Function	Use for	Switch	Mode	Function	Use for					
TG1	IU			PSF	M	SetShiftButton (UMD - U)		APAT	U							
	OU	Primary Fire	PWS		M				M	fnSetJoystickCurves(NONE)						
	IM				M	SetShiftButton (UMD - M)			D							
	OM	fnAdvFireControl(0)	Normal	PSM	M			APAH	U							
	ID	tgTriggerMode(1)			M	SetShiftButton (UMD - D)			M	fnSetJoystickCurves(MILD)						
TG2	OD	fnAdvFireControl(1)	Disco/Mining	PSB	M	SetShiftButton (UMD - D)		APALT	U							
	U				M				M	fnSetJoystickCurves(MEDIUM)						
	M				D				D							
S1	U	tgHUDMode()		SC	U	ShowConnectionStatus		EAON	U							
	M	tgHardPoints()			M	ResetHeadOrientation			M	tgSilentRunning(ON)						
	D	SwapRollYaw			D				D							
S2	IU			LTB	U	tgExtCamera()		EACOFF	U							
	OU	Secondary Fire	PWS		M	LookOnOff			M	tgSilentRunning(OFF)						
	IM				D				U							
	OM	fnAdvFireControl(2)	Normal	CSU	U			RDRNRM	M	tgCargoScoop(DEPLOY)						
	ID	tgTriggerMode(2)			M	tgWarpDrive(0)	Supercruise		D							
S3	OD	fnAdvFireControl(3)	Disco/Mining	CSD	U			RDRDIS	U							
	U				M	tgWarpDrive(1)	Hyperjump		M	tgCargoScoop(RETRACT)						
	M	SetShiftButton (IO - I)			D				D							
S4	D			CSR	U			LDGH	I	mClearAllChatBox						
	M	tgEnhancedFAOFF()			M	SelectNextSystemInRoute			O	mClearChatBox						
	D				U											
H1U	U	tgFSSMode()		MSU	I			APENG	I	fnHangerServices(1)						
	M	NextFireGroup			O	Alternate Roll Control (CCW)			O	fnRequestDock()						
	D	DSSModeOFF														
H1D	U			MSD	I			FLAPU-R	U	fnVoiceVolume(INCREASE)						
	M	PrevFireGroup			O	Alternate Roll Control (CW)			M	Galaxy Map	ON					
	D				I				D							
H1L	U			MSL	O	Alternate Yaw Control (Left)		FLAPM	U							
	M	tgLandingGear()	tempo						M	Galaxy Map	OFF					
H1R	U	tgLights(1)	Night Vision	MSR	I			FLAPD	U	fnVoiceVolume(DECREASE)						
	M	tgLights(0)	tempo		O	Alternate Yaw Control (Right)			M	SystemMap	ON					
H2U	D				I				D	SetLED OFF						
	M	SelectTargetAhead		MSP	O				U	fnVoiceVolume(DECREASE)						
	D	SelectTargetAhead						FLAPD-R	M	SystemMap	OFF					
H2D	U			SPDF	U	tgReverseThrust(0)	No Speedbrake		D	SetLED ON						
	M	SelectHighestThreat			M	tgReverseThrust(1)	Speedbrake	APUON	U							
H2L	D	WingmanTarget			D	tgReverseThrust(2)	fix LED		M	fnDeploySRV(DEPLOY)						
	M	SelectNextShip		SPDF-R		tgReverseThrust(1)	Forward		D							
H2R	D	SelectWingMan1, 2, 3			U			APUOFF	U							
	M	SelectNextHostileShip		SPDB	M	EngineBoost			M	fnDeploySRV(RECOVER)						
H3U	D	WingManNavlock			D				D							
	U			BSF	U	mNAVBeaconWingON/OFF		EORIGN	U							
	M	fnPIPManager(ENG)			M	tgText2Speech()	EnableVoice ON/OFF		M	fnDRShip()	Dismiss/Recover					
H3D	D			BSM	D				D							
	U				U			EORNORM	U							
H3L	M	BalancePower		BSB	M				M							
	D				D			EORMOTOR	U							
H3R	U			CHF	U	mReportCrimesToggle			M	HideShipGUI	GUI OFF					
	M	fnPIPManager(SYS)			M				D							
H4U	D			CHB	D			EORMOTOR-R	U							
	U	fnAdvancedSCB(2)	2xSCB, 1xHS		U	TargetNextSubsystem				M	HideShipGUI	GUI ON				
H4D	M	fnAdvancedSCB(0)/(1)	SCB, SCB+HS		M	TargetPrevSubsystem			EOLNORM	U						
	D	CameraZoomIN			D					M	fnModeSwitch(0)	Open				
H4L	U			EOLNORM	U					M	fnModeSwitch(1)	Private Group				
	M	fnHeatsink()			M					D	fnModeSwitch(2)	Solo				
H4L-R	D	CameraZoomOUT			D			EFROVER	U							
	U	ChargeECM (DOWN)	Charge ECM	EFLNORM	U				M							
H4R	M	ChargeECM (UP)	Fire ECM		M	TargetPrevSubsystem			D	GameLoaded = 0						
	D	O			D			EFLOVER	U							
H4R	U	fnChaff()		IDLEON	M				M	fnSetLEDs()	Reset Status LEDs					
	M	CameraBlurIN			D				D							
H4P	U	fnPIPMode(0)	Reset	IDLEOFF	U			EFLOVER	U	fnDebugStates(DebugON)						
	M	fnPIPMode(1)	Increment		M	fnPIPMode(2)	Decrement		M	fnDebugStates(DebugOFF)						
H4P	D	fnPIPMode(2)			D				D							
	U				U				U							
H4P	M	fnHangerServices(1)			M			IDLEOFF	M	fnHangerServices(0)	Launch					
	D	fnHangerServices(2)			D				D							

HOTAS Switch and Button Reference – BASIC

HOTAS - Quick reference - v4.2.0 - BASIC											
Joystick				Throttle (lever)				Throttle (base)			
Switch	Mode	Function	Use for	Switch	Mode	Function	Use for	Switch	Mode	Function	Use for
TG1		Primary Fire		PSF				APAT			
TG2				PSM				APAH			
S1		Hardpoints		PSB				APALT			
S2		Secondary Fire		SC		ResetHeadOrientation		EACON		Silent Running	
S3				LTB		LookOnOff		EACOFF			
S4		FA-Off		CSU		Supercruise		RDRNRM		Cargo Scoop	
H1U		NextFireGroup		CSD		Hyperjump		RDRDIS			
H1D		PrevFireGroup		CSL		SelectNextSystemInRoute		LDGH			
H1L		Landing Gear		CSR		OrbitLines		APENG			
H1R		Lights		MSU		Alternate Roll Control (CCW)	(no rudders)	FLAPU		Galaxy Map	
H2U		SelectTargetAhead		MSD		Alternate Roll Control (CW)	(no rudders)	FLAPD		SystemMap	
H2D		TargetHighestThreat		MSL		Alternate Yaw Control (Left)	(no rudders)	APUON			
H2L		TargetNextShip		MSR		Alternate Yaw Control (Right)	(no rudders)	APUOFF			
H2R		TargetNextHostileShip		MSP				EORIGN			
H3U		PIPs - ENG		SPDF		Reverse Thrust		EORMOTOR		HideShipGUI	
H3D		PIPs - Balance		SPDF-R		Forward Thrust		EOLIGN			
H3L		PIPs - SYS		SPDB		EngineBoost		EOLMOTOR			
H3R		PIPs - ENG		BSF				EFRNORM			
H4U		Shield Cell Bank		BSB				EFROVER			
H4D		Heatsink		CHF		TargetNextSubsystem		EFLNORM			
H4L		ECM		CHB		TargetPrevSubsystem		EFLOVER			
H4R		Chaff						IDLELON			
H4P								IDLEOFF			

Disclaimer

This software is a work in progress.

No warranties are stated or implied and no responsibility will be accepted for any damage this software may cause to your hardware, sanity or marital status.

By downloading, installing and using this software, you agree this is at your own risk.