

KEYBINDING  
BEST PRACTICES  
AND  
PHILOSOPHY

FOR HOTAS, HOSAS,  
AND OTHER HANDS-ON ALPHABETS

COMPILED BY  
CHANTICROW

LAST UPDATED: 22-MAY-2022  
V1.8

# Table of Contents

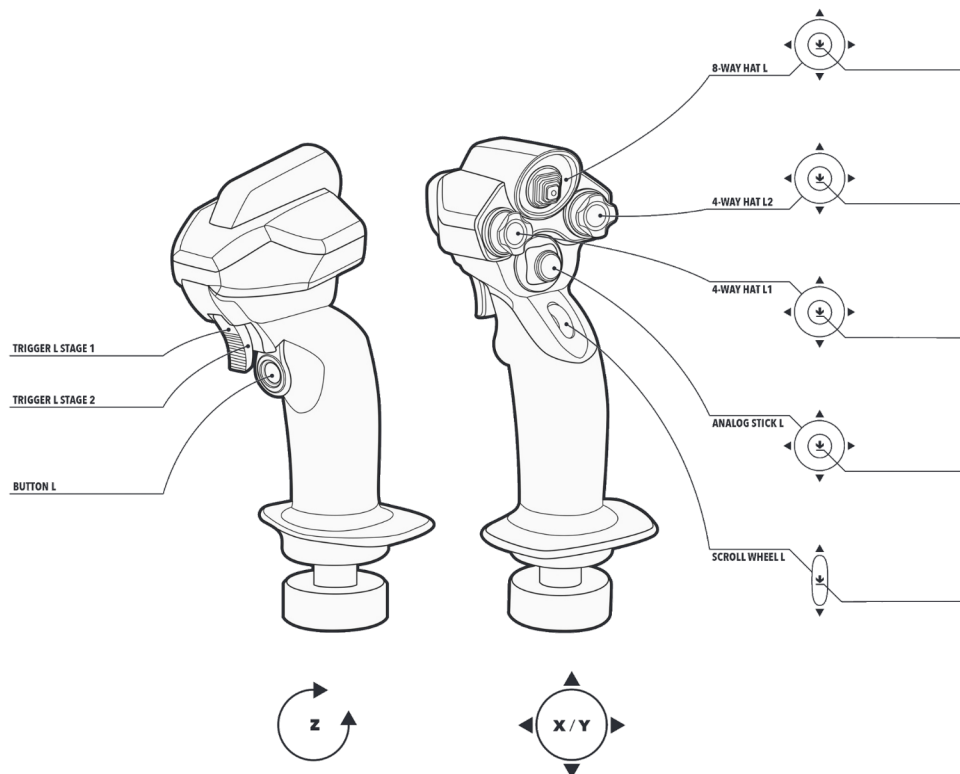
Who is this for?	4
This document is...	4
Feedback	4
Priority Binding Tiers	5
Tier One - Movement, Self Preservation, and Offense	5
Tier Two - Targeting, Communications, and Fleet Defense	6
Tier Three - Power, Engineering, and Navigation	6
Tier Four - Everything Else	7
Control Zones and Ease of Access	8
Best Practices	9
Documentation	10
Taking It Further	10
Ergonomics	11
Thumb Training	12
Resources	13
Tutorials	13
3rd Party Programs	14
Hardware Mounts	14

# Credits

This document would not be possible without the contributions of the following:

TheN00bifier  
KEMI65  
/u/Auctoria\_RK1

/u/GentleFoxes  
/u/Milk\_A\_Pikachu  
Foxx Mounts



Flight hardware illustrations are the copyright of the respective hardware manufacturers. Virpil Delta illustration enhanced by /u/yogidragon.

# Who is this for?

This document is aimed at space and fictional setting flight sim pilots seeking to optimize the layout of bindings on sticks and throttles. Some concepts may be applicable to mouse and keyboard and game controllers as well.

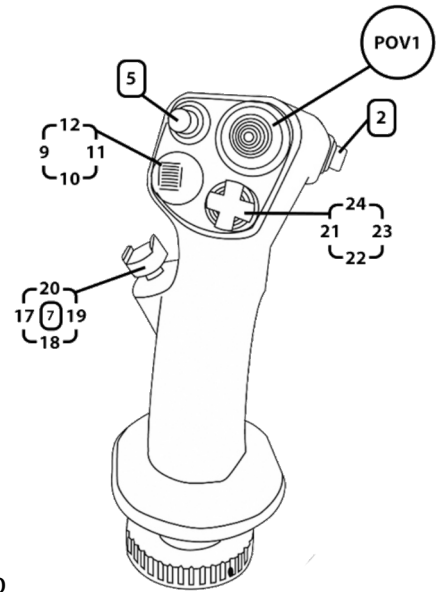
If you are a real-world flight sim enthusiast looking for the most accurate reproduction of a specific plane's hardware, then this document is not for you.

Every pilot's bindings should be organized to make sense for the pilot. If none of the suggestions in this document make sense to you, ignore them and do your own thing.

## This document is...

- ...a generalized guide for deciding where to bind functions on home flight hardware.
- ...a set of opinions from the experience of one sim pilot.
- ...not a specific set of directions on where to bind functions.
- ...not a guide for a specific game's bindings.
- ...not a recommendation for a specific combination of hardware devices.

This document started while collecting my thoughts around how to optimize my bindings, and as a repository for "best practices" I learned while studying the bindings of other pilots. The priority tiers were my way to codify my thoughts on which functions to bind first, second, and so on.



## Feedback

Feedback and suggestions for new best practices are welcome.

Look for Chanticrow in the HOTAS/HOSAS/SIMPIT Discord at <https://discord.gg/szqaJE7>.

# Priority Binding Tiers

The goal is to make the most often used and critical functions as accessible as possible.

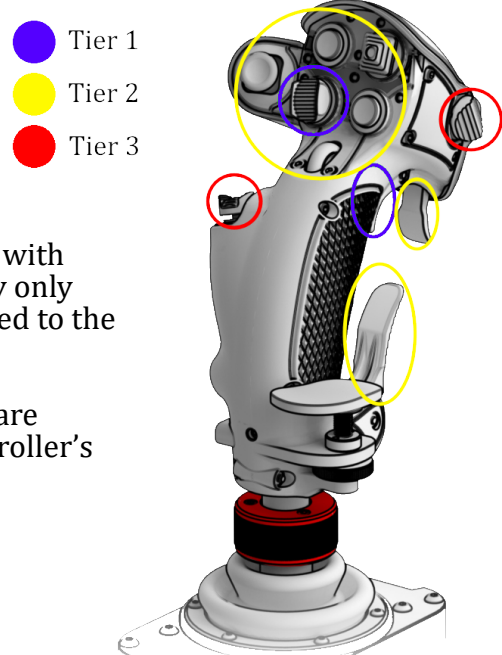
Presented below are four tiers of functions. The higher the tier, the more time critical the function.

Functions mentioned in each tier are examples, not a definitive list. Games may prioritize different game play elements during high stress situations.

For example, if the game makes scanning cargo containers during combat an important aspect of play, make that function a tier one item. If your game requires frequent and immediate communication with other players then comms may be a tier one function. Adjust as necessary.

Prioritizing functions is crucial when binding to hardware with fewer buttons. The Thrustmaster T16000m has effectively only eight buttons on the stick (not including the base) compared to the thirty plus buttons on the Virpil Alpha.

Ultimately, it is up to the player to decide which functions are critical and which to relegate to the wilderness of the controller's base.



## Tier One - Movement, Self Preservation, and Offense

These are the most used or most critical actions. These should be bound on controls where your fingers rest (see [Thumb Training](#) below) and can act with the least amount of response lag.

- **Movement and speed management** - Afterburner, decoupled/gliding, braking, speed and throttle limiters, matching speed, and special automatic movements such as looping or flipping 180 degrees.
- **Threat assessment and hostile targeting** - Nearest hostile or attacker, cycling hostiles, reticle focus, missile and incoming threat detection, target of target.
- **Weapon firing and selection** - Firing weapon groups, switching weapon groups, reloading weapons.
- **Countermeasures** - Any countermeasure and decoy selection and deployment.
- **System management** - Specifically items for self preservation such as shield power boosts, shield balancing, and coolant flushes.
- **Special Actions** - Any special offensive or defensive deployments that could turn a defeat into a survivable situation. This does not include ejection.

## Tier Two - Targeting, Communications, and Fleet Defense

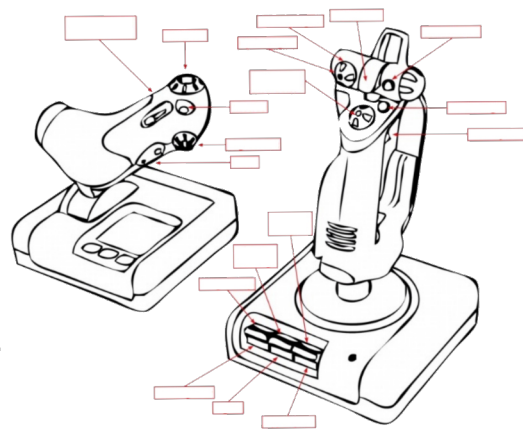
These controls should surround the Tier One functions. They are still frequently used, but need less critical reaction time to engage.

- **Communications** - Real world comms for multiplayer games. Push-to-talk, comm channel switching.
- **Targeting** - Subsystems, friendlies, containers, target pinning and recall, deselecting targets.
- **Wingman Management** - Any comms and targeting related to interacting with and defending your wingmen.
- **Radar Management** - Any radar management functions that may impact the ability to find hostiles or complete objectives within a time limit.
- **Shield Management** - Shield facing adjustments.
- **Power Management** - Any power management functions lending directly to offense or defense, such as all power to shields or weapons.

## Tier Three - Power, Engineering, and Navigation

Map these functions to any remaining controls available on the hardware's grip, and then to controls on the hardware base. These functions are suited to harder to reach buttons and controls using shift modifiers.

- **Shift Modifier** (3rd party tools) - Shift modifiers should only be used for less critical support functions and for accessing deeper functionality within non-combat systems. Using a shift modifier during battle is prone to inaccurate function selection and requires more time to access the functions behind the modifier.
- **Menu Selection** - Some games, such as Elite Dangerous and House of the Dying Sun, offer menu interaction to control many functions of the ship or deal with tactical decisions. Depending on the game, this may be a tier two function.
- **Communications** - Any remaining NPC communication and management functions.
- **Power Management** - Any remaining power management functions.
- **Navigation** - Accessing maps, setting routes and waypoints.
- **Scanning** - Deep target scanning, exploration scanning.
- **Engineering** - Functions dealing with the repair or modification of the ship in flight.
- **Faster-than-light Travel** - Activating FTL, hyperspace, quantum, jump drives, etc.

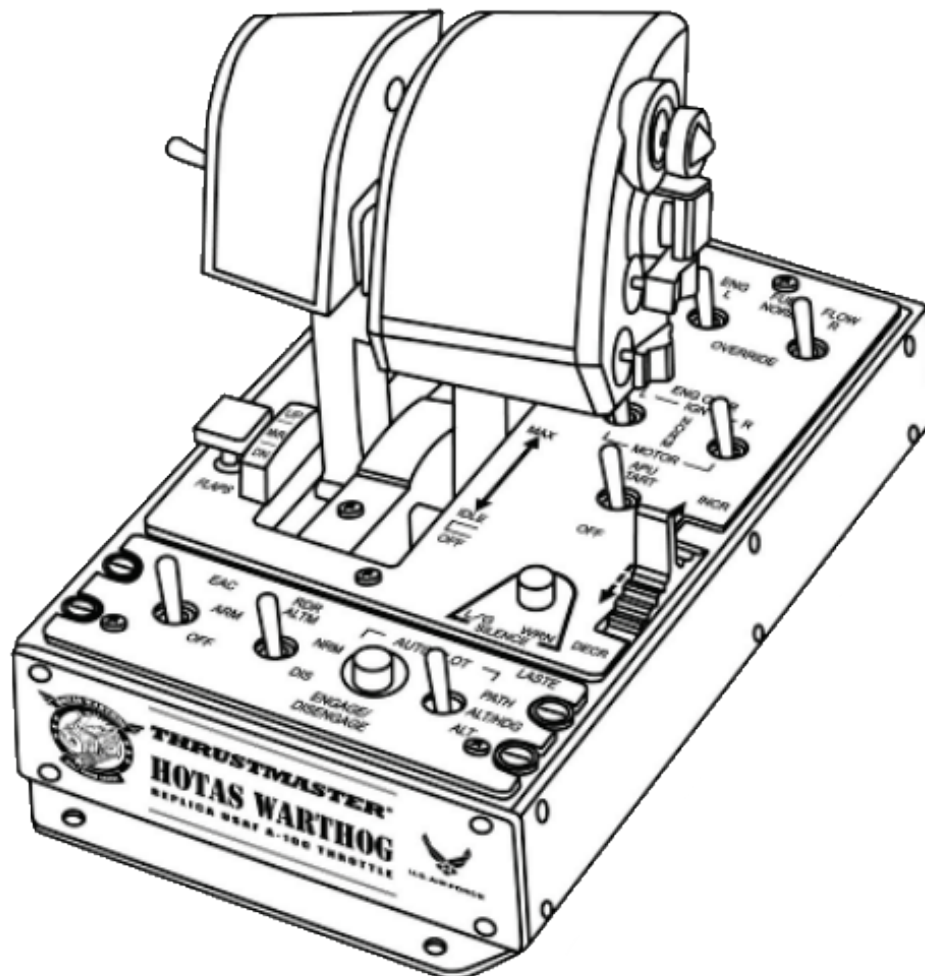


## Tier Four - Everything Else

Any functions not included in the above lists go here, or are assigned to the other tiers at the user's discretion. This includes items such as lights, landing gear, door controls, non-combat modes such as mining or exploration, etc.

When you get to this point, remember that not every function needs binding. Sticks and throttles are hands-on so you can focus on the most critical and time sensitive functions. The more functionality crammed onto the hardware, the more likely it will create confusion or lead to accidental activations of the wrong functions.

Games such as DCS and Star Citizen have a huge variety of functions, many of which are used infrequently. Take advantage of keyboard bindings and virtualized cockpits in games that offer them for these seldom activated functions.



# Control Zones and Ease of Access

Sticks and throttles have groups of controls organized for a specific digit of the hand. This document will refer to these control groups as zones.

Most frequently for sticks is a thumb zone and an index finger zone. Throttles often have zones for all digits. In most every case, the thumb zone presents the largest grouping of controls.

Some devices offer multiple zones for the same digit. These are mutually exclusive where that digit must rest in one zone or another, and cannot access controls in both zones at the same time.

The mutual exclusivity of zones and the number of controls in a zone should be considered when assigning bindings.

Functions that need to be accessed at the same time need to be assigned in zones that are not mutually exclusive, or that command less of a finger's attention. For example, one may need to activate push-to-talk comms while also dogfighting and selecting targets. The function for push-to-talk should go on one hand while targeting is assigned on the other hand.

The finger zones usually offer fewer controls to deal with at one time. Assign functions to these controls that may need activation while the thumb is engaging other functions. Examples are afterburners, weapon firing, and braking.

If using a 3rd party program, such as Joystick Gremlin, that allows the use of modifier keys, the modifier should be on a finger that is less likely to be engaged in with other controls. In most cases, the index finger zones or pinky finger zones are better selections for the modifier. The thumb zones will always be the worst option because they access the most controls at once. Placing the modifier on an index or pinky finger allows the most combinations with the thumb zoned controls. HOTAS users should put the modifier on the stick, not the throttle, since the throttle hand will most often be used to interact with the keyboard, and the throttle usually has a wider variety of inputs for modification.



Zones on the Virpil Alpha  
Note the mutually exclusive thumb and finger zones.



Zones on the Thrustmaster T1600m

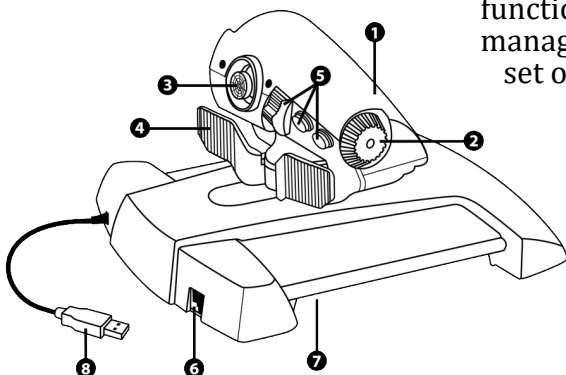


# Best Practices

The following items are additional practices to consider while setting up bindings

- **Do** bind weapon firing on the opposite hand from weapon aiming. This prevents the action of pulling the trigger from creating tremors in the aim. Professional photographers use a remote shutter trigger for this same reason.
- **Do** bind functions associated to directions to controls matching those directions. For example, braking might be mapped to a control that pulls backward or down.
- **Do** use common conventions such as up/forward as “on” and down/backward as “off”.
- **Do** bind controls to match visual interfaces. For example, shield facing distribution using four faces might be bound to a hat switch corresponding to the four facings. A hat switch may be mapped to match a power management screen layout with Engines/Shields/Weapons. An MFD or button panel may be mapped to closely match the on-screen cockpit layout.
- **Do** bind to controls purpose built for a function, when possible. Dials for zoom functions or trim wheels. Toggles for power and lights. Hats for direction based functions. The more naturally a control fits with the function, the easier it will be to remember and use.
- **Do** bind the same functions on the same controls across games, as much as possible. Binding landing gear, hostile targeting, weapons activation, etc on the same controller input regardless of game allows muscle memory to work across games. If you have the inputs to spare, do not bind other functions to these designated controls. If the game does not offer landing gear, do not bind a different function to the button usually used for landing gear.
- **Do** determine if any hat switches are easier to push or pull based on your hand position. Push is often the easier and faster action. When binding to the hat switch, bind more time sensitive items to the direction that is easier for your thumb to access.
- **Do not** bind frequently accessed or critical functions behind a shift modifier. It always takes longer to press two buttons in sequence than it does to press a single button.
- **Do not** bind eject or self destruct to the stick or throttle grip.
- **Do not** bind major condition shifts such as power on/off, VTOL, mode changes, and landing gear deployment on controls that are easy to hit accidentally.
- Take advantage of mutually exclusive modes of operation to repurpose controls. For example, in Star Citizen the Flight Mode, Mining Mode, and Scanning Mode are mutually exclusive. Therefore, the weapon group fire, mining laser and extraction mode switch, and the scanning activation and scanning ping can all be assigned to the same buttons.

**System vs Task Binding:** Real cockpits are designed with controls for a specific system clustered together so the brain always seeks the same area for a specific functionality. If your hardware has sufficient buttons, hats, and switches you might also bind functions related to a single system, such as power management, shield facings, or targeting, to a single control or set of adjacent controls.



If your hardware is light on inputs, consider binding by task instead. For example, on a single hat you might bind “nearest hostile”, “cycle hostiles”, “match speed”, and “wingman, attack my target”, all functions related to the task of setting up your next enemy.

# Documentation

It is important to make a binding map for later reference. Search online for blank keybind maps of your hardware. Fill these maps with your bindings, and keep them up to date.

Make notes about which bindings you modify in game. Changing a binding in game is a fleeting moment in time, and will be forgotten by the next time it needs to be changed. Write it down in case the game controls need to be rebound again later.

Make backups of your binding files. Most games and 3rd party programs save bindings in an external file. Keep these backed up with your other off-site backups, or face the agony of rebinding all those controls after that unexpected hard drive crash.

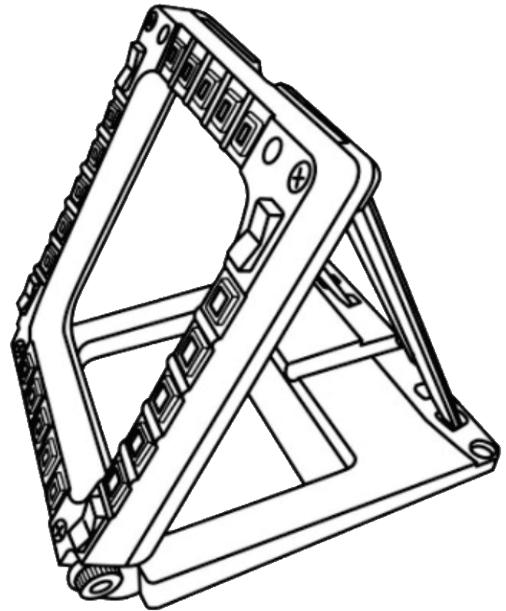
## Taking It Further

Using 3rd party programs can allow many shortcuts and macros to push the bounds of bindings. They can also make it possible to use modern stick and throttle hardware in older games with less developed hardware compatibility.

Some examples are:

- Automatically activating afterburners when the throttle or stick reaches the edge of the throw distance.
- Conditional activations such as firing one weapon group only if another group is not already firing.
- Short and long press of controls to activate different functions.
- Combining or splitting of axes.
- Advanced axis curves and axis inversion.
- Shift modifiers to allow double or triple function assignment on a single control.
- Binding hardware controls to keyboard key presses.
- Scripting for more advanced concepts. For example, there is a script for Joystick Gremlin that combines dual sticks into tank/skid controls.

The ability to bind hardware controls to key presses instead of binding directly to in-game functions can be a time saver for games in development, or for users that frequently uninstall/reinstall games. For example, binding hardware via Joystick Gremlin to a game's default bindings allows an easy backup of those bindings outside the game. If something wipes the game's bindings, the process to reset the bindings is minimal because most of the mapping is already done in Joystick Gremlin.



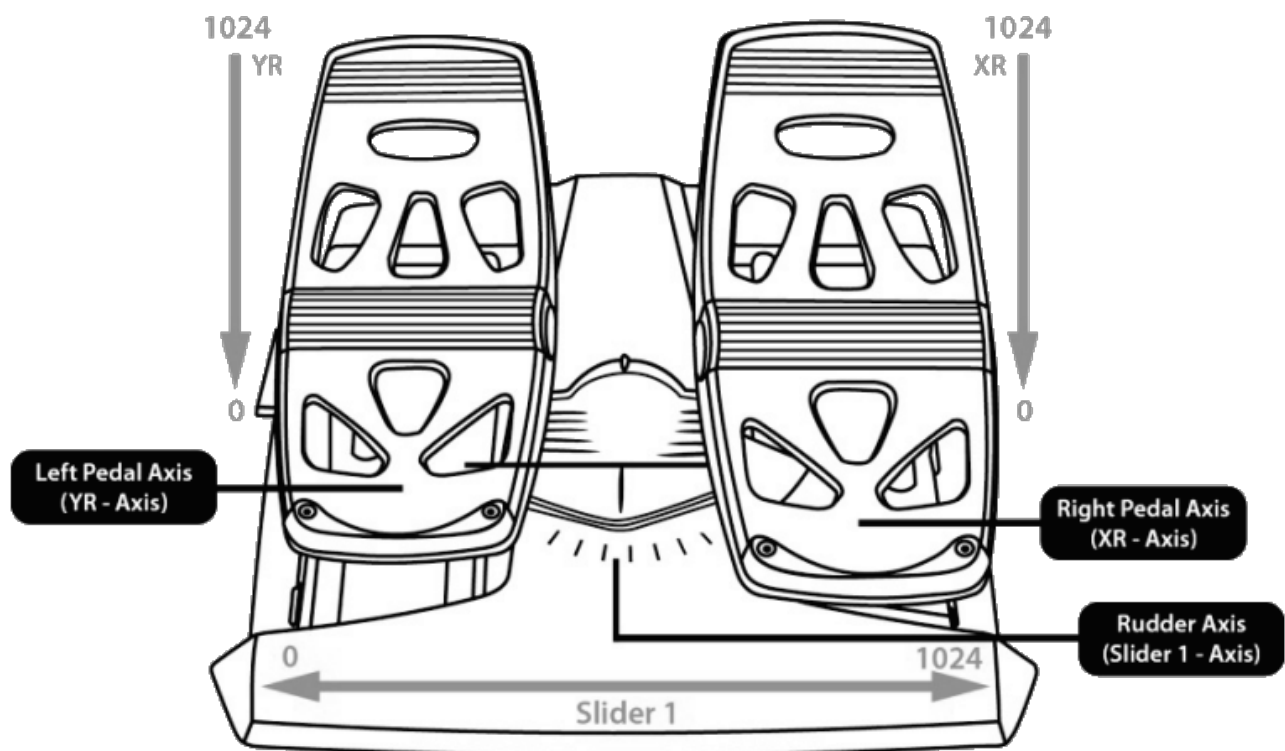
# Ergonomics

Correct positioning of hardware is crucial to avoid bodily strain and damage. Improper hardware positioning can lead to nerve compression problems, such as carpal tunnel, or worse.

Start by sitting with your arms straight down. Bend your elbows so your forearms are parallel to the floor. In this position your hand should be able to rest comfortably on the stick's hand rest or on top of the throttle's grip. The shoulders should droop, and not need to be held up while gripping the controls. The hand and wrist on the throttle should not have to rotate in order to rest on the grip.

The shoulder, arm, and wrist should be relaxed and at neutral angles while on the hardware. If this is not the case, such as when the hardware is sitting on top of one's desk, then a mounting solution may be needed.

A variety of mounts are available that allow the hardware to be raised, lowered, angled, and rotated as necessary to reach an ergonomic solution.



# Thumb Training

Be sure to practice proper thumb placement to keep those carefully mapped controls available at a moment's notice.

The thumb should always rest in the thumb zone with the most controls. If the thumb is resting away from the controls this creates a lag time in your response to changing conditions during flight. It is natural for the thumb to rest in a relaxed side position, but this position is often not optimal for control access. The thumb tends to move back to this incorrect resting position during lulls in high stress situations.

Be mindful and train yourself to keep your thumb in the correct position while flying.



Correct Thumb Positions



Incorrect Thumb Positions



# Resources

## Tutorials

A variety of sites and videos for guidance and education regarding flight hardware and bindings.

### HOTAS/HOSAS/SIMPIT Discord

By far the most useful hangout to chat with other flight hardware users.

<https://discord.gg/szqaJE7>

### BuzZzKiller's Dual Joystick Setup Series

The later portions of this guide are Star Citizen specific, but the "Why Use Dual Joysticks" video is an amazing overview of various hardware setups and their advantages.

<https://www.youtube.com/playlist?list=PLcfoKKM84nAV4JceoPGAqmPXcBx10y0Wx>

### N00bifier's Virpil Alpha Star Citizen Bindings

Binding examples for the Virpil Alpha.

[https://www.youtube.com/watch?v=1-x\\_gGh42L0](https://www.youtube.com/watch?v=1-x_gGh42L0)

### Ventorvar's Joystick Configuration Tutorial

A complete walkthrough on how to set up Joystick Gremlin. Includes recommendations for axis and curve configurations.

<https://ventorvar.com/posts/joystick-intro/>

### Havoc Company's Everything Virpil Playlist

Questions about anything Virpil related? Havoc Company has the answer.

<https://www.youtube.com/playlist?list=PL2kRNYjsz2IDZWueRjHKHMqHvpYJnA8wD>

### Val's "Best Control Method"

This video is also Star Citizen specific, but the concepts apply to any flight sim that provides six degrees of freedom. Val does an incredible job of explaining coupled versus decoupled, gimbaled versus fixed weapons, degrees of freedom, and how various controller setups work with these concepts.

<https://youtu.be/XeQFqWdAmsw?t=278>

### Afterimage Flight's Dual Stick Tips and Tricks

JaguarMG demonstrates his amazing simpit and explains his keybinding setup for Star Citizen.

<https://www.youtube.com/watch?v=EXwLC2blftg>

### RedLir's Joystick Gremlin Guide

<https://youtube.com/playlist?list=PLorfWq-uIpBFF15CjrHtnwMqYDRrDr7ZA>

## 3rd Party Programs

A list of recommended software that allows expanded functionality for hardware beyond what most games offer.

[Joystick Gremlin](#)

[vJoy](#)

[JoyToKey](#)

[Universal Control Remapper](#)

[HIDHide](#)

[DevReorder](#) - May not work with anti-cheat software.

[Joystick Diagrams](#)

## Hardware Mounts

[Predator Mounts](#) - US

[Foxx Mounts](#) - US

[Monstertech](#) - US

[Monstertech](#) - UK

[Pein Gear](#) - South Korea

[DX Mounts](#) - UK

