VPforce RHINO FFB

User Manual

Version 0.90 (WIP)

2025-07-01

Table of Contents

1. Welcome

Whether you just opened a box of Rhinos, are waiting for yours to arrive or just want to know more about the beast, you have come to the right place. This manual is intended to provide you with all the relevant information on what the Rhino is, what it is capable of and how you can get the most out of one.

First of all, the Rhino is huge. When it comes to sheer size and weight, most other stick bases aren't even close. The dimensions are available in this manual, so that you can prepare beforehand - and it is highly recommended that you do so. The shape is simple enough so that you can even build a cardboard model to assist in figuring out a suitable mounting for your setup. Also note that due to the long distance from the gimbals to the stick mounting point, the Rhino basically comes with a built-in extension and at a maximum of 22 degrees each way, the stick throw is significant out of the box.

Mounting is not critical just because of the weight of the Rhino, but also its strength. At a maximum of 9 N·m of Torque the Rhino can provide about 3+kg of pull with typical stick setups. Compared to most springs & cams -controllers, this is about the maximum they can do with the heaviest springs, but they really only give you that much resistance at the very edges of avia-style cams, whereas the Rhino can be more linear. In practice, the Rhino is probably the strongest controller you have used, even if it's not exactly in a class of its own.

Another key characteristic of the Rhino is its flexibility. Whereas with many other controllers adjusting the basic qualities such as the amount of resistance or the shape it comes in can make your fingers bleed, with the Rhino you can simply adjust a few sliders in the software. You can also mount pretty much any grip you wish to use - Thrustmaster and Virpil directly, VKB and Winwing with an adapter.

All the flexibility comes with a price, however. The Rhino does anti-cogging, natural damping compensation and a number of other forms of black magic to make the operation smooth and seamless. On simpler, closed systems such as the venerable Microsoft Force Feedback 2, everything is optimized for the one, simple configuration they come with, ensuring smooth operation straight out of the box. When you can mount for example either a Virpil

MongoosT-50 grip or an extended Warthog with five times the weight and maybe twice the height, there is no one size fits all -settings to ensure perfect operation for both. Although the Rhino should be fine with default settings, some amount of adjustments will be required to get the most out of any specific setup. No need to worry about it, though, that's exactly what this manual and the accompanying video guides are for.

With a tool as powerful as the Rhino, the results of less than optimal settings or a misbehaving game can be somewhat spectacular, which is why the system comes with a big red button that makes everything instantly safe again. One more reason to pay attention to the mounting arrangement, make sure you have easy access to all the controls. Despite its strength, encounters with an angry Rhino usually don't end up with lethal consequences and sometimes even catastrophic injuries can be avoided. Mostly the Rhino just wants to slap your fingers a bit or play a friendly little game of WF[1], so no need to worry too much.

Last but not least, different games and simulators offer a whole new level of complexity. Some, such as Il-2 Great Battles and DCS support DirectInput FFB right out of the box, others (MSFS, XP) need external software. There is also an ongoing telemetry project for richer FFB in DCS. Luckily and as usual, this manual contains a lot of the info you need to get well on your way with at least the most popular simulators out there - and towards what for many of you out there will be a paradigm shift in controlling virtual planes, spaceships or whatever floats your boat.

[1] https://www.youtube.com/shorts/2XXK-gT5Sko

2. Getting Started

Force feedback in general is a somewhat complicated topic, largely thanks to all the possibilities the technology provides. Getting the most out of your VPforce Rhino will require some setup and adjustment, but thankfully you can eat Rhinos just like you eat elephants: one bite at a time.

This section of the manual includes all the relevant information to get the Rhino running in a basic configuration, both the physical configuration and the very basics of the software. Once you are done with this section you should have a functional game controller you can start using right away, while you dive deeper into the technology in the following sections.

2.1. Technical Specifications

Weight: 5.2 kg

Size: 205mm x 180mm base,
height to the top of the grip connector 290mm

Motors:

- Two type 57BLF03 NEMA servo motors
- Max 30A drive current per motor
- Resolution: 14 bit / Rev, 13 bit effective stick resolution
  12 bit / Rev, 11bit effective stick resolution (before 2023 Q2)
- Typical power: 150W

Maximum torque: 9 N·m per-axis

Gimbal: 3D Printed PETG / Aluminum / Bearings

Transmission: 1:6.2 ratio belt drive

Maximum throw: 22 degrees each direction

Built-in functions: 1 bindable rotary axis (defaults to spring force strength)
1 emergency button that cuts off all power to the motors

Cooling: Two fans kick in when the motors reach 50°C

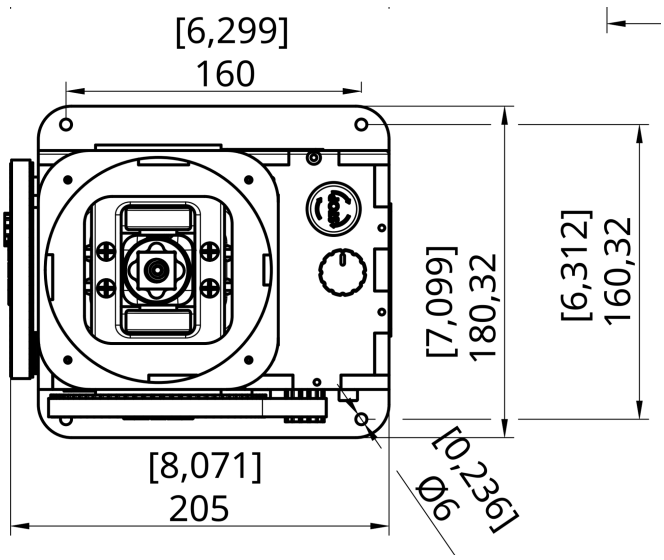
Power source: 180W, 20V

Grip Compatibility:

- Thrustmaster directly
- Virpil directly (T-50CM2, WarBRD, Constellation Alpha, V.F.X)
- VKB with an adapter and a black box
- WinWing with an adapter

2.2. Physical Setup

The VPforce Rhino can be attached to any suitably durable and rigid mounting with four 6mm bolts (see schematic).



Due to the weight and strength of the hardware, the mounting structure needs to be designed accordingly. Also note that the gear and belt drives are on the outside of the base structure and it is important to make sure that they do not make any contact with other parts of the mounting system or the person in the pilot's seat.

Attach the power cable and the included USB cable to the Rhino base and then connect them to the power socket and computer respectively.

The big red button is an important safety feature, make sure that you have unrestricted access to it at all times. To facilitate this you can rotate the Rhino and reverse the axis in software (see 2.4 for details).

To cut off all power to the motors, simply push down on the big red button until it locks.

To reset, rotate clockwise and let the button pop up.

The rotating knob can be used to control different features of the Rhino (see software sections for details), by default it controls the spring force.

The Rhino supports Thrustmaster (Cougar, Warthog, F/A-18C) and Virpil grips directly.

Note that rotating Thrustmaster grips requires the use of an extension. For VKB and Winwing see below.

2.2.1. The VKB Adapter

The VKB Adapter allows for the mounting of any socket B - style VKB grips. In addition to the Adapter, you will need a main controller (black box) to operate VKB grips with the Rhino. Note that the adapter has an external cable that attaches to the black box (see picture).

To connect the adapter to your VKB grip, push the connector into the grip until it makes contact and secure with the little screw thing - basically exactly as with a normal VKB base.

Important: if the connector is tight and doesn't want to go in easily, DO NOT apply force to the rotating lower part. This can pinch and mangle the wire coming out of the adapter. Sitting on the adapter is also highly not recommended.

There is no electrical connection between the adapter and the Rhino, so you can simply place the adapter on the Rhino connector, rotate the grip freely into a suitably ergonomic position and screw the adapter's lower part in until the connection is secure enough to hold the grip without rotating under stress. Re-tighten if necessary.

Note: The black box will blink a red light, because it doesn't see any axis (only the grip connects to the black box). This is normal and will not hamper the operation of the device.

To use a VKB grip's buttons for force trim or other functions in the Rhino software, you need to use the included RhinoLoopback app. See details in the relevant section.

2.2.2. The WinWing Adapter

The WinWing adapter adapts the WinWing grips to the Rhino interface both mechanically and electrically. Tested and correctly working correctly with the WinWing F-16EX and F-18 grips.

It converts the proprietary WinWing protocol to a TM standard 5-pin interface and also passes analog axis data such as brake lever and thumbsticks. It should be compatible with TM/Virpil bases, but without the analog axis functionality and possibly limited to 24 buttons.

In some rare cases a WinWing grip will not report analog axis data, in that case a calibration of the grip analog functions needs to be performed via WinWing software on a WinWing base.

To use the full grip functionality on the Rhino base the “WinWing adapter” grip type needs to be selected in the drop down menu.

On newer revisions of the WinWing adapter firmware, button number 32 will illuminate if the Grip connection with the WinWing grip is not functioning/disconnected.

2.2.3. The RHINO Throw Limiter Adapters

The RHINO has a significant amount of throw - 22 degrees - and a long shaft compared to most comparable controllers. The long throw does help with accuracy, but especially with extensions the wide movement range can become excessive. It is possible to set limiters in software, but depending on how much force you are using in general, the software limiter may not feel strong enough. The physical throw limiter adapters offer a simple to install alternative solution that sets hard physical limits to the stick’s movement.

The adapters come in two pieces - front and back. They can be ordered in different configurations and may have different movement ranges in different directions. To install the adapter plates,

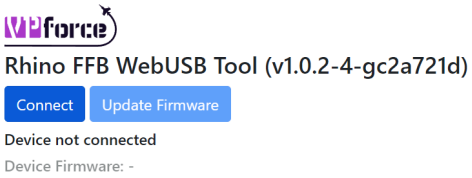
1) Unscrew the four Torx T5 screws that connect the dust cover at the base of the stick shaft to the RHINO base and lift the cover slightly - you don’t need to remove it completely. With the cover out of the way, you should now be able to see the opening the stick shaft goes through and the top of the gimbal assembly inside the base.

2) Insert the two limiter plates in the stick shaft opening. **Text side is up** and **FWD is forwards**. When installed correctly, the plates should fit snugly in the opening and stay firmly in place.
3) Place the dust cover plate on top of the limiter plates so that the screw holes align and reattach the screws so that they hold the whole shebang in place. There is no need to go full gorilla on the screws, but do note that they are now holding in place not just the dust cover, but also the limiters that make physical and sometimes somewhat forcible contact with the stick shaft.
4) Recalibrate the controller for the new throw ranges.

2.3. Initial Connection and Updating the Firmware

As you connect the Rhino to the computer for the first time, it should give you a popup with a link to the VPforce WebUSB tool website. If it doesn't or you miss the popup, the site is https://vpforcecontrols.com/usb/rhino/

Clicking "Connect" on the website to gain access to Firmware update and some basic configuration utilities. Click "Connect", choose the correct stick (for most it's the only Rhino on the list) and "Connect".



If the system detects that a newer firmware is available, it will offer you the choice of updating it - unsurprisingly known as "Update Firmware".

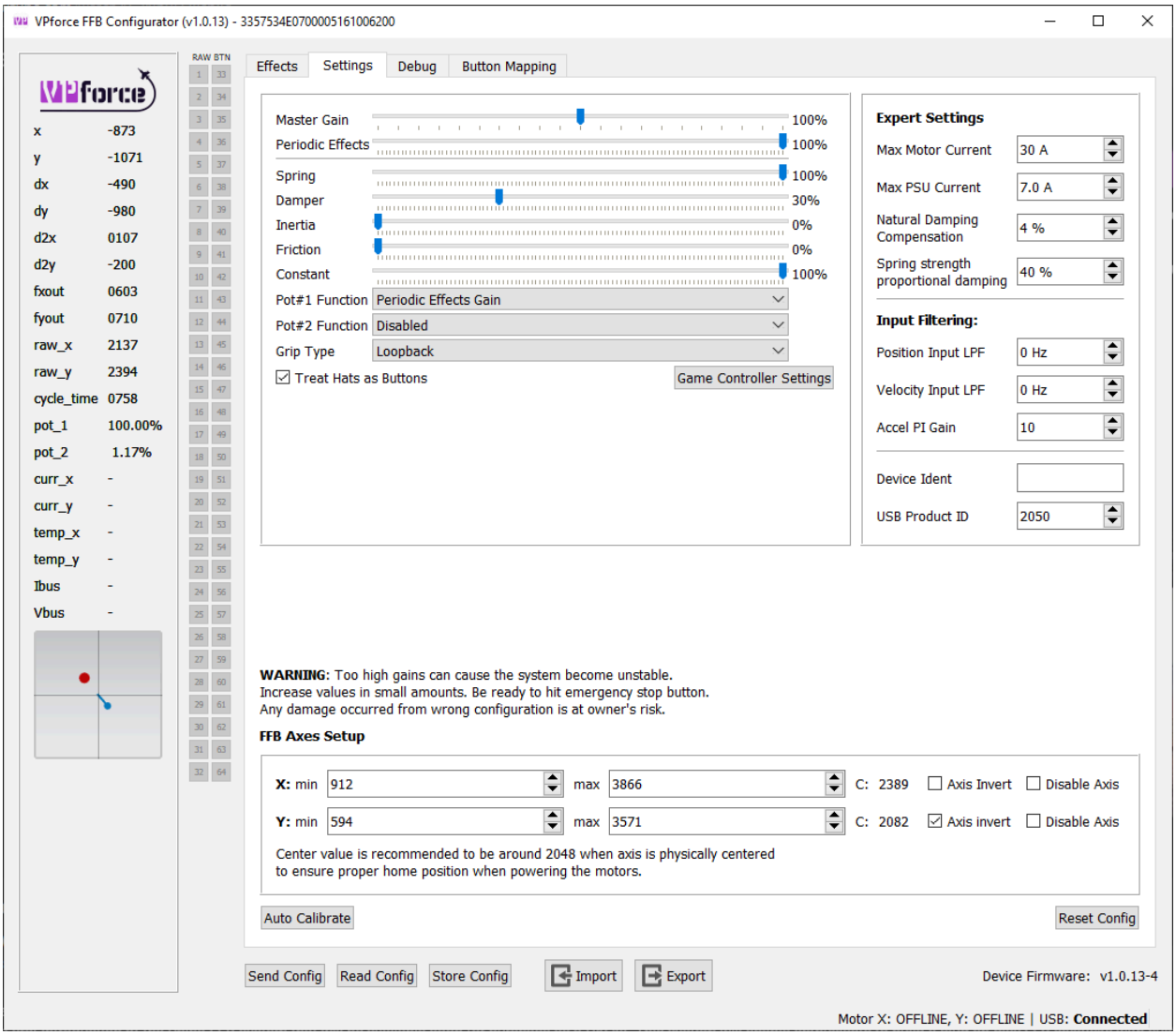
On the same website you can also download the Rhino Desktop Software package.

Note: You must be running a chromium based browser in order for the WebUSB functionality to work (Google Chrome, Brave, Microsoft Edge, Opera, etc..). Firefox does not support WebUSB

2.4. Installing the Software and Basic Operation

First go to the https://vpforcecontrols.com/usb/rhino/ website and download the software package. The latest one is recommended, although older ones are also available. The software comes in a simple zip package, unpack it to a location of your choice. To start the configuration software, double click “VPforce_FFB_Configurator.exe” located in the folder you just created.

You should now be greeted by something that looks a bit like this:



You should be able to see raw_x, raw_y values react to the stick position and also curr_x, curr_y along with some other values. If you don’t (like the picture above), check that the power

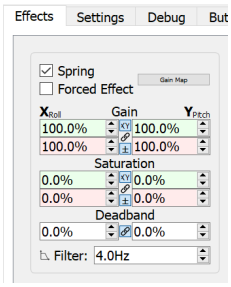
supply and USB cable are connected and the big red button is not pushed in (rotate clockwise and let it pop up if it is). More troubleshooting help is available in the relevant sections.

The first thing you need to do is to calibrate the sensors. Click “Auto Calibrate” in the Settings tab and move the stick to all the extremes. Next turn off “Auto Calibrate” and click “Send Config” below. If you want to save the calibration more permanently, click “Store Config”.

- **Send Config** makes the current configuration active, but does not store it to the device, so it gets lost during restarts.
- **Store Config** stores the configuration permanently to the device’s flash memory.
- **Read Config** reads the configuration stored in the device and adjusts the software setting accordingly.

Tip: If the maximum movement range feels excessive, see “Endstops” in the main software section.

Once the calibration is complete, you are almost ready to go and understanding just a couple of the settings should get you on your way. First of all, “Master Gain” in the “Settings” tab is basically your main power setting. Set it to 100% to find out what the RHINO is capable of or use more modest settings to suit your specific setup. “Periodic effects” (below Master Gain) are basically stall shake, gun fire and other effects produced by equivalent events in games.



If you want to get a feel for the controller right away, go to the “Effects”-tab and make sure “Spring” is selected and gains are at 100%. Also make sure that the “Spring” slider in the “Settings”-tab is not in the zero position. Due to how force feedback works, the stick will be basically limp unless there is input from a game or other software. The “Spring” option in the FFB Configurator will provide basic centering so that you can get a feel for the RHINO and play with the other settings. Choosing “Forced Effect” is not recommended at this point, because it will keep the Configurator spring force always active and prevent other software (like your favorite simulator) from taking over properly when it needs to.

For more details see the main sections on the configurator software and game specific settings, but you should now be ready to start your adventures in learning the way of the force feedback.

3. Using the RHINO

The VPforce RHINO is a powerful tool and power usually comes with an inevitable degree of complexity. RHINO is not an exception and combined with the plethora of existing force feedback implementations in different simulators as well as the 3rd party software required for some of them - not to dismiss the enhanced DCS exports project by VPforce itself - the situation can be quite complicated, indeed. Fortunately the amount of complexity is perfectly manageable with a little bit of help and this document is specifically designed to provide that help.

First of all, the RHINO should work pretty well right out of the box. Of course it's tempting to immediately dive into all the options that can make the RHINO experience just a little bit better, but my advice is to actually try your favorite sim and get some actual stick time before tinkering with every setting. Especially if you aren't used to Force Feedback - like many probably aren't nowadays - the experience will help understand what all the adjustments actually do. It will also help to have some fun in between settings diving sessions so that you don't drown in the ocean of options.

This document provides instructions for the main FFB Configurator software, the RHINO Loopback app, basics of *TelemFFB* - which is intended to provide more advanced FFB effects for DCS World - as well as instructions on how to best utilize force feedback and the RHINO in different simulators. Also included is a basic description of what force feedback is, how it works and what is the functionality behind the terminology.

3.1. Overview and Force Feedback Terminology

3.1.1. FFB Overview

Force Feedback (Control loading) is a technology used to provide users with a more realistic and immersive experience in virtual environments. It involves the application of physical resistance to input devices, such as joysticks, pedals, or steering wheels, to simulate the feeling of controlling a real-world device.

Force Feedback can be used to simulate the resistance of mechanical systems, such as the control surfaces of an aircraft, the suspension of a car, or the movements of a robot arm.

By providing users with tactile feedback that corresponds to their actions in a virtual environment, Force Feedback enhances the realism and engagement of their interactions, making it easier to learn and practice complex tasks.

Force Feedback can be combined with other technologies, such as motion tracking, to create a more convincing and effective virtual experience for users in a wide range of applications, from aviation and automotive training to entertainment and gaming.

3.1.2. FFB Effect types

Periodic Effects: These effects generate a repeating waveform, which is modulated in amplitude, frequency, or phase. The waveform can be sinusoidal, triangular, or square, and the modulation parameters determine the specific sensations felt by the user. Periodic effects are used to simulate sensations such as vibrations, oscillations, and pulses.

Spring: This effect provides a linear restoring force that is proportional to the displacement of the input device. The spring constant determines the strength of the force, and the effect can be used to simulate the feeling of a mechanical spring, such as the resistance felt when pushing down on a button or pulling on a joystick.

Damper: This effect provides a damping force that is proportional to the velocity of the input device. The damping coefficient determines the strength of the force, and the effect can be used to simulate the feeling of a viscous fluid.

Inertia: This effect provides an inertial force that is proportional to the acceleration of the input device. The mass parameter determines the strength of the force, and the effect can be used to simulate the feeling of a heavy object moving or the sensation of acceleration.

Friction: This effect provides a static or dynamic frictional force that is proportional to the displacement or velocity of the input device. The friction coefficient determines the strength of the force, and the effect can be used to simulate the feeling of different surfaces, such as a slippery road or a sticky track.

Constant: This effect generates a constant force in any direction. But it can also be modulated, i.e. updated by software dynamically thus achieving any desirable effect from host software.

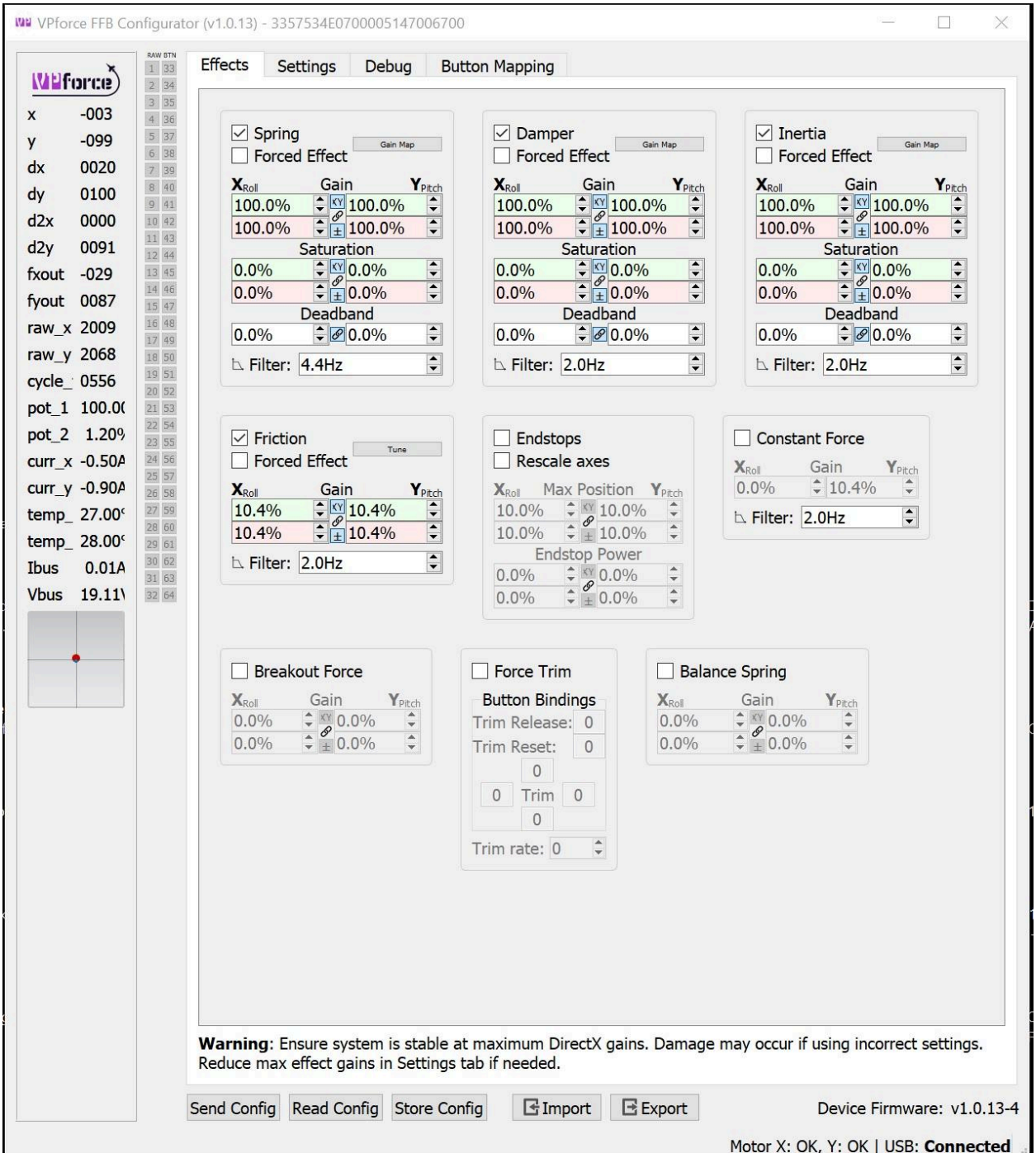
3.2. The FFB Configurator Software

VP Configurator

The left pane is a list of variables affecting the behavior of the Rhino. It is a real time display of the values sent from the joystick.

The right pane consists of 4 tabs where the behavior of the joystick can be modified.

3.2.1. Effects tab



If a checkbox is checked, then that function is active.

Functions definitions:

Spring effect

a force is generated by the motors in the control unit that feels like spring which return the joystick to center.

Forced effect:

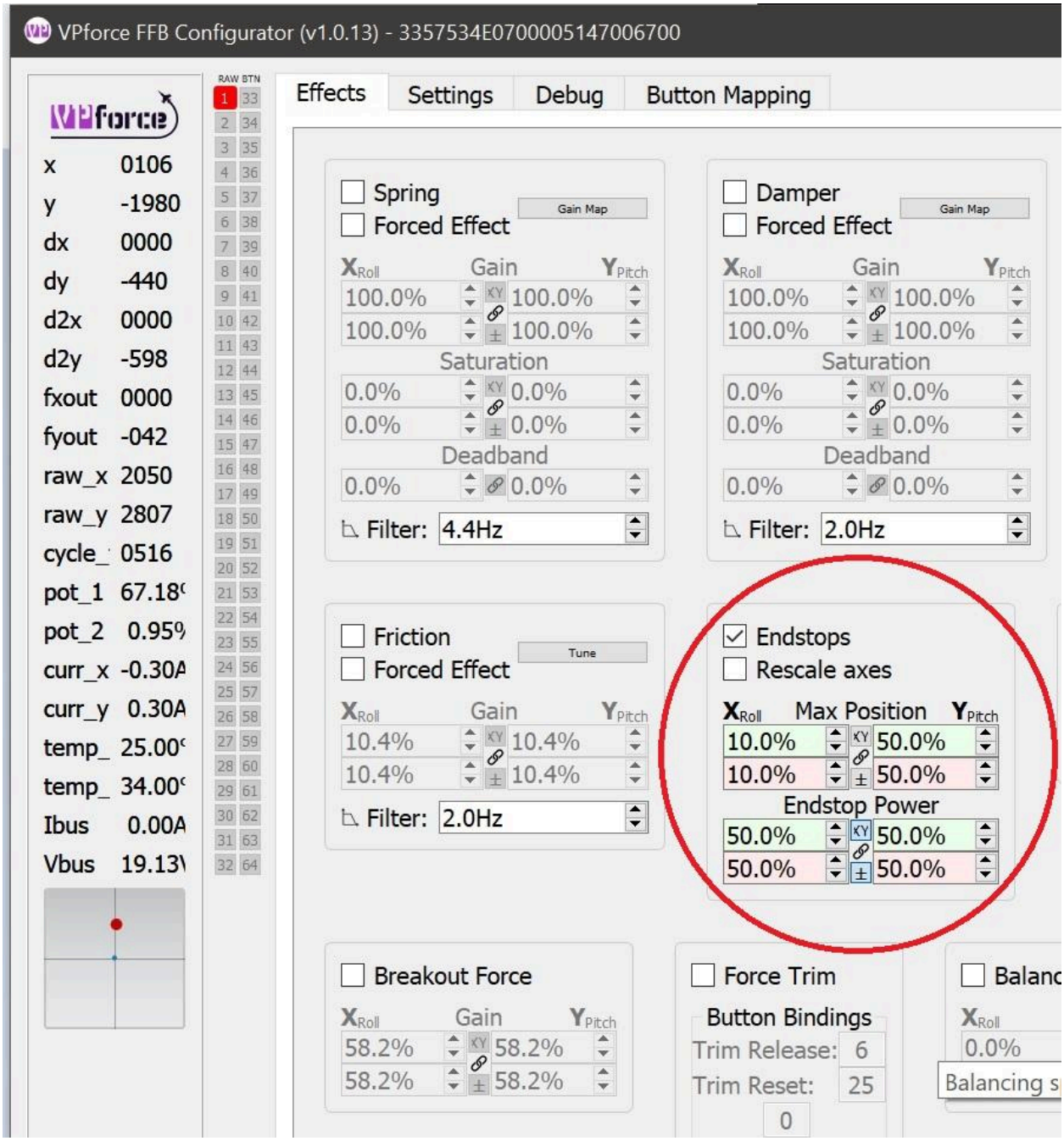
Gain:

Saturation: the percent of the maximum force

Deadband: the region near the center where no force is active

Filter:

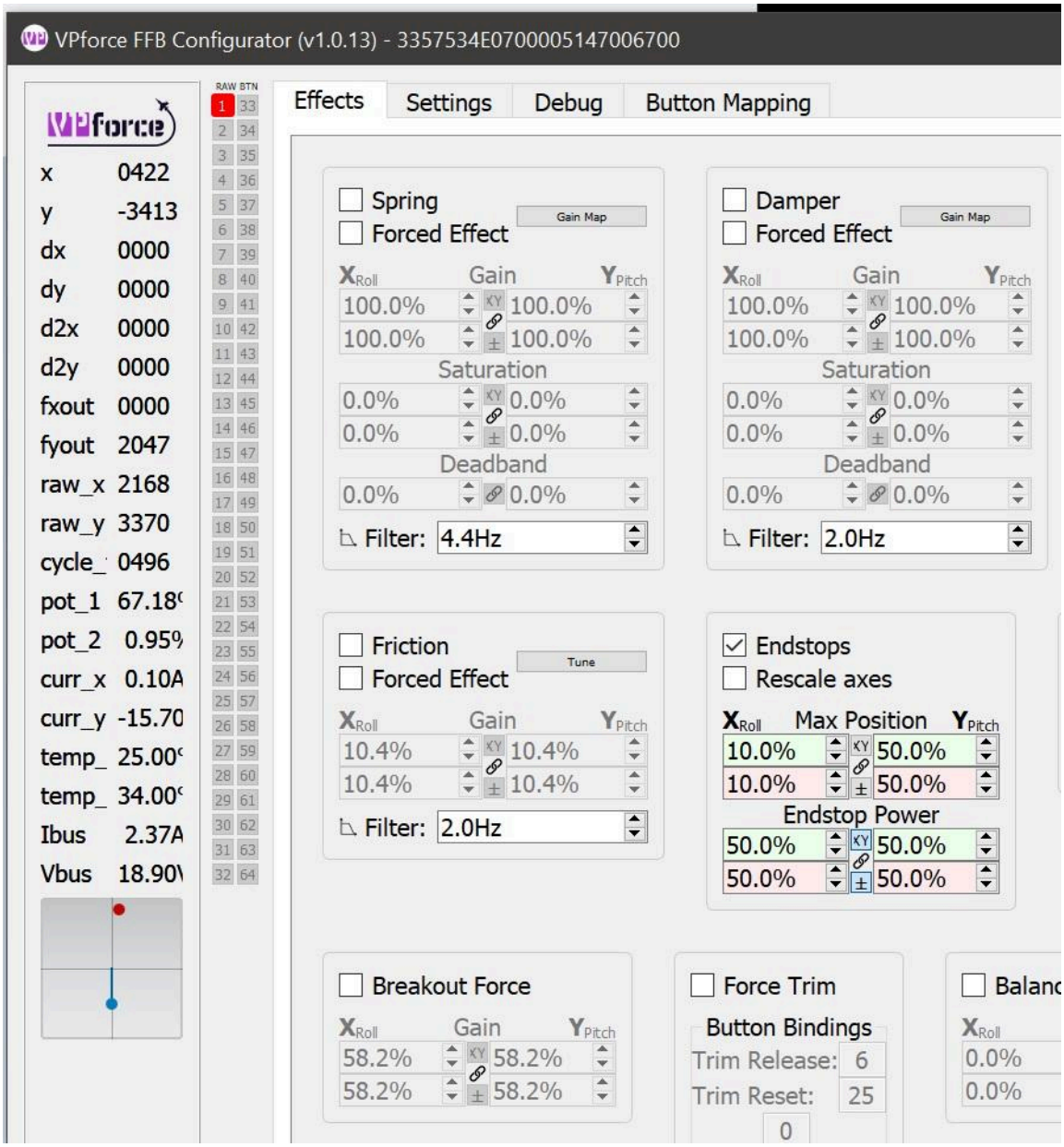
- **Damper effect**
  A force is generated by the motors in the control unit that feels like the joystick is moved through a viscous fluid. The resistance increases the faster the joystick is moved.
- **Inertia effect**
  A force is generated by the motors in the control unit that feels like the joystick has momentum and tends to continue moving in the direction it was moved.
- **Friction effect**
  A force is generated by the motors in the control unit that feels like the joystick is moved through a fixed resistance.
- **Endstops**
  A force is generated by the motors in the control unit when the stick reaches a certain point. These parameters are set in the green and red colored boxes in the region of the page titled ‘End Stops’.



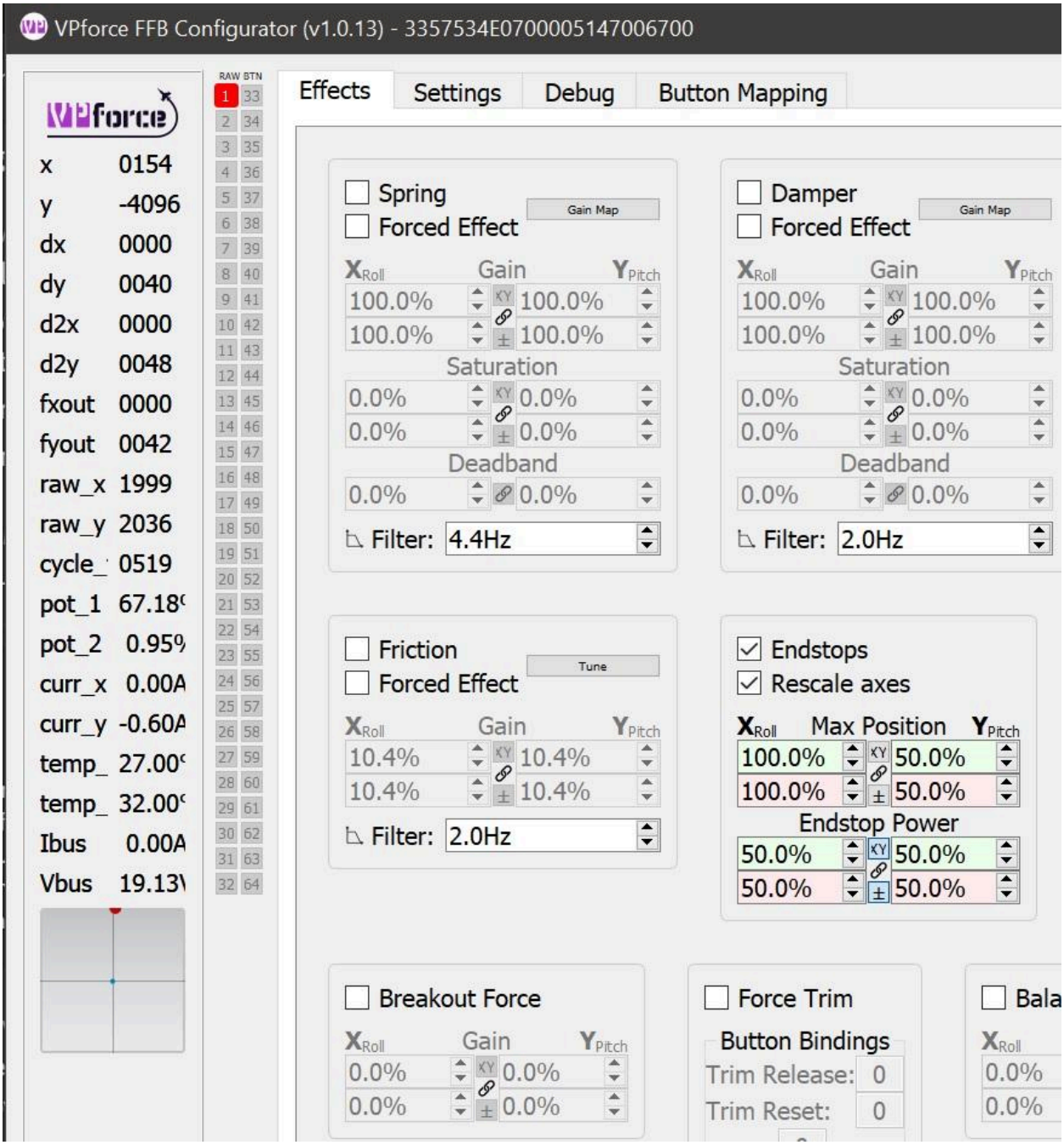
In the picture above, endstops have been set at 50% Y travel in both directions with 50% maximal force to be generated at each endstop. Each direction is contained in a different colored box.

For Y, green is the direction of pushing the stick away from you. Red is the direction of pulling the stick toward you. This configuration results in the stick moving freely fore and aft about halfway in each direction without resistance. It is entirely limp. However, at the halfway

excursion, stiff resistance is encountered. If you look at the little graph in the bottom left of the picture, you'll see the red dot is halfway forward and the blue dot is still at center. The red dot indicates stick displacement. The blue dot is the vector of the force applied by the motors. If the stick is pushed further through the resistance, the red dot is seen to advance to the limit and the motors are pushing against you with half of their maximal force. This is depicted below.



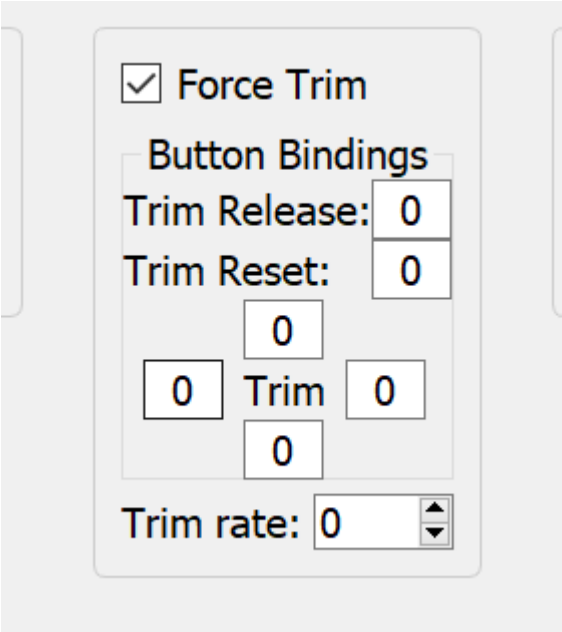
If the rescale axes box is checked, then the joystick will tell the computer it has reached full excursion when it reaches an endstop. Below you can see this as the red dot has moved to the limit of the little box. However, the blue dot is still centered because the motors are not doing anything.



Constant Force

Breakout force

Force trim

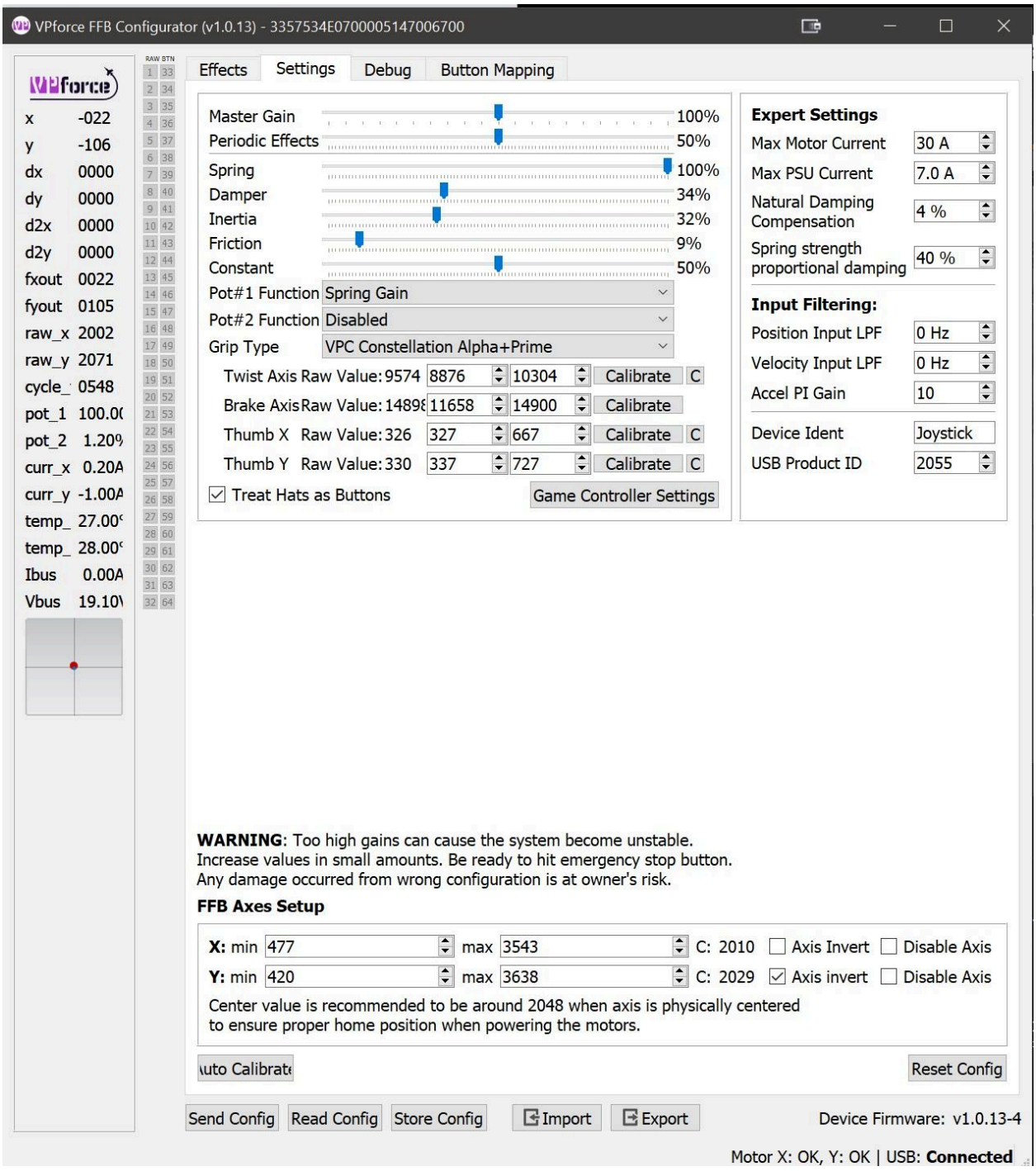


This setting allows a trim button and a trim release for helicopters

The four fields surrounding the word ‘Trim’ are button assignments for pitch and roll trim.

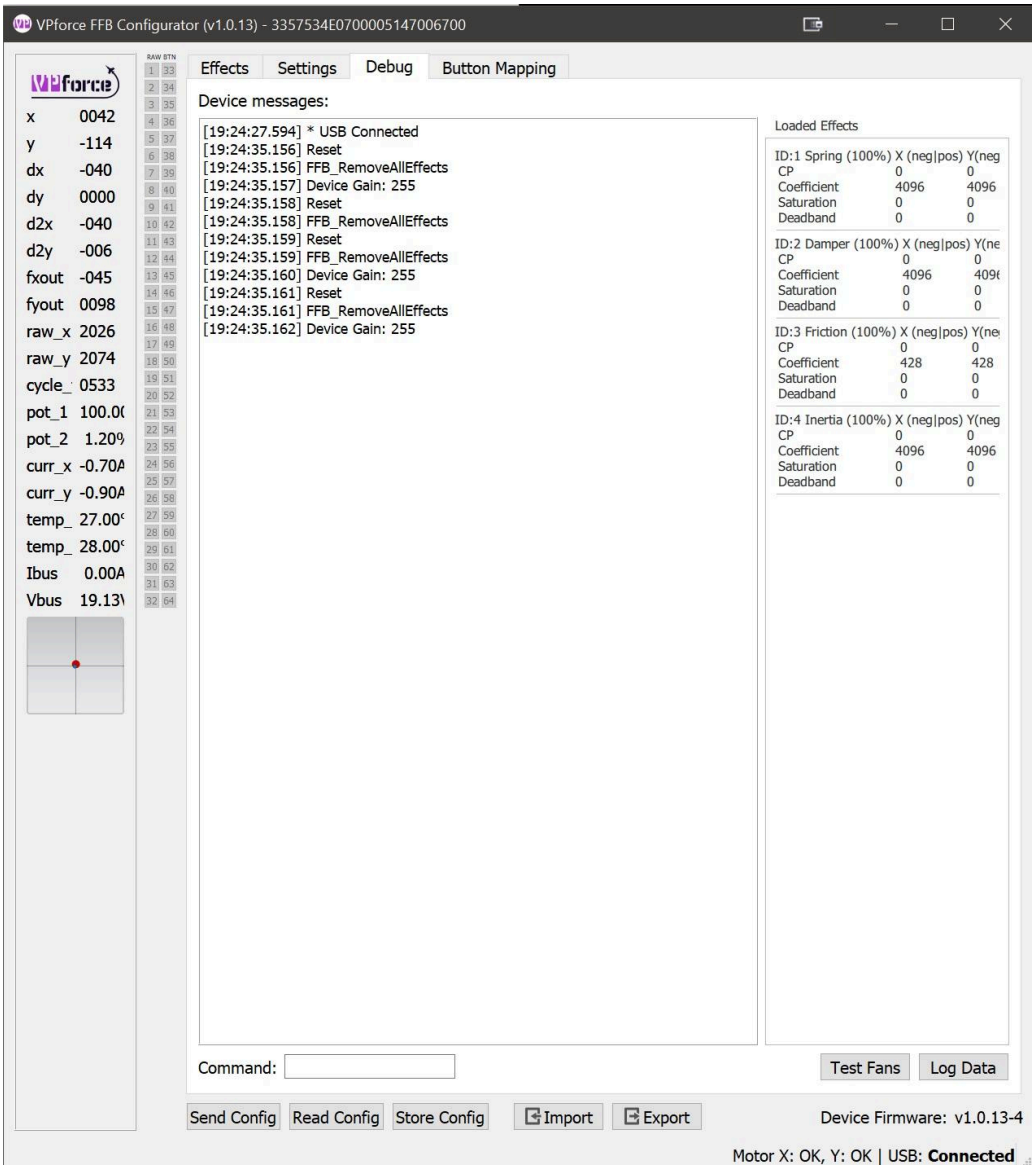
Balance spring

3.2.2. Settings tab

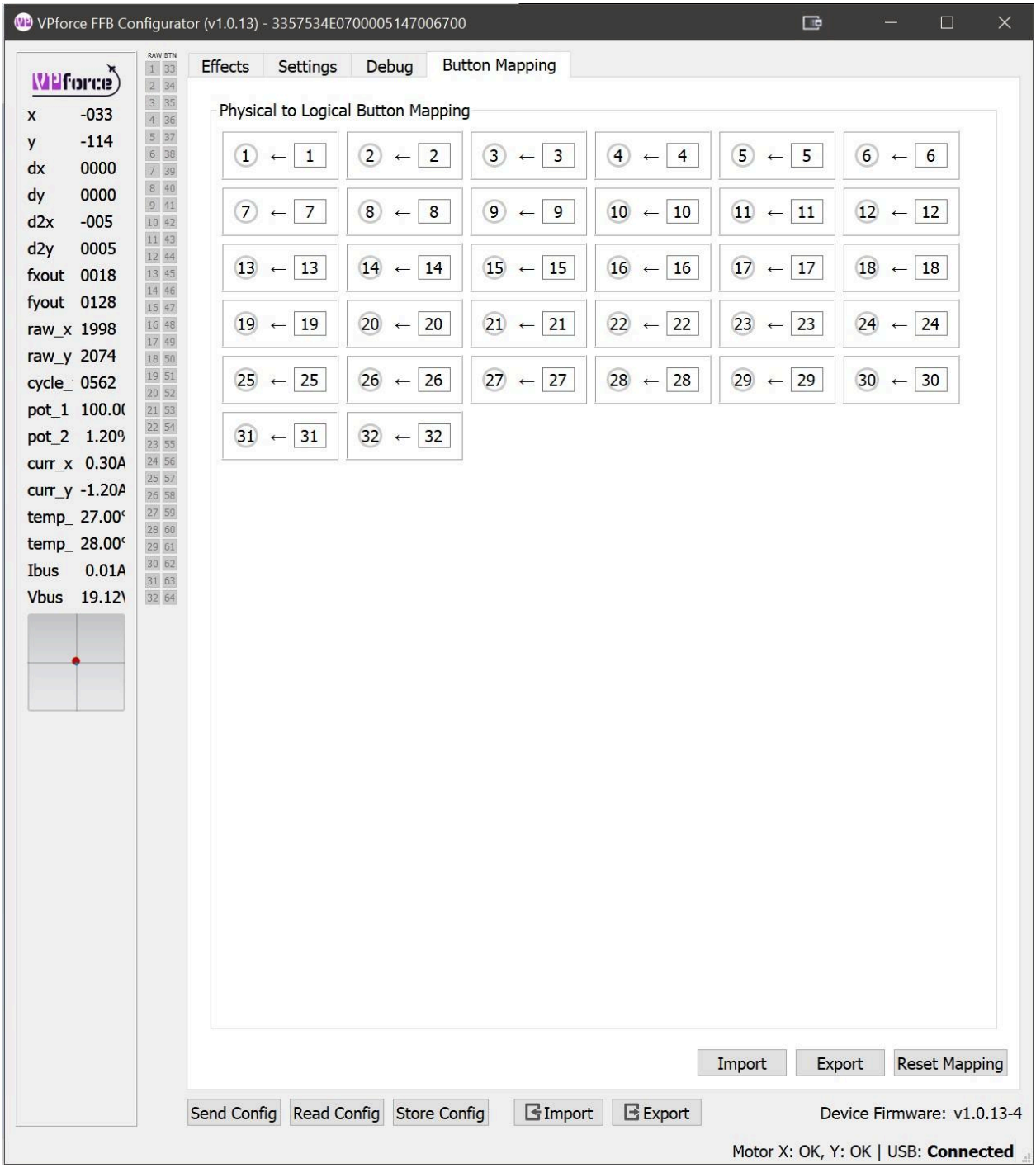


Each slider allows amplification or reduction of the effect set in the previous Effects tab. The total effect felt by the user will be the product of these two numbers. So for example if the spring effect is set to 50% in the effects tab, and the slider on the settings tab for spring is set to 50%, then the user will feel 25% of the maximal force. There is a reason for providing these two controls separately. (I dont know what the reason is but Walmis certainly does)

3.2.3. Debug tab



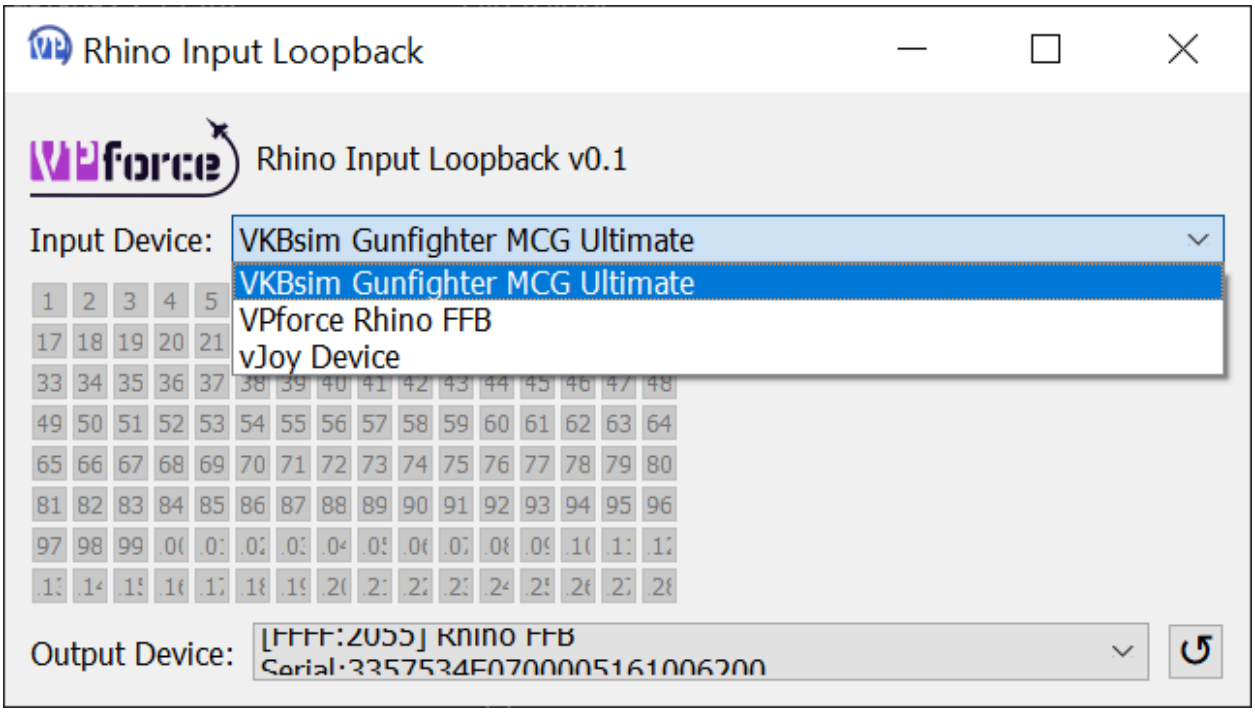
Button mapping tab



3.3. The RhinoLoopback Application

The purpose of the RhinoLoopback app is simply to allow for using any controller - including VKB installed on a Rhino - to control VPforce FFB Configurator features, such as for example the Force Trim (see relevant section in 3.2.).

To start the application, navigate to the directory you unzipped the software package (VPforce_FFB_Configurator_vx.x.xx.zip), locate Rhinoloopback.exe and start it. You should see something like this



Choose the input device you want to use from the list (in this case the VKBsim Gunfighter MCG Ultimate), Output Device will in most cases be the only Rhino on the list. Note that the **RhinoLoopback app needs to be running for the loopback to work**. To finalize the operation, go to the FFB Configurator software settings tab and set Loopback as the Grip Type.

Configuring the RhinoLoopback app in this way will now allow the main FFB Configurator to see, map and utilize the buttons of the selected device.

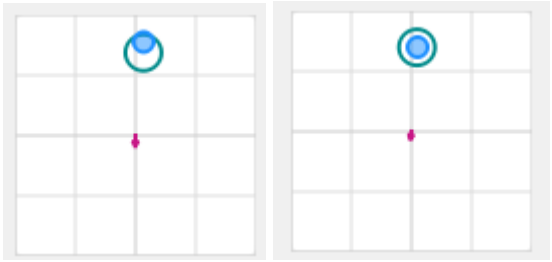
3.4. Balancing the Grip

3.4.1. Balance Spring

The "Balance Spring" feature, found within the VP Configurator, allows users to achieve optimal trim performance, particularly when using heavier grips or extensions. This manual entry provides instructions on how to tune the "*Balance Spring*" settings effectively.

The usual problem with a heavy grip and/or a long extension, is that it will sag or drift when trimmed at an angle as indicated in the picture. This occurs because additional torque is required to hold the stick in place at an angle. The spring force is proportional to the stick distance from the center point, therefore the stick falls a bit until countering torque is achieved, which is undesirable for force trim applications. This effect is most noticeable with a weak spring setting.

Before balance: After balance:



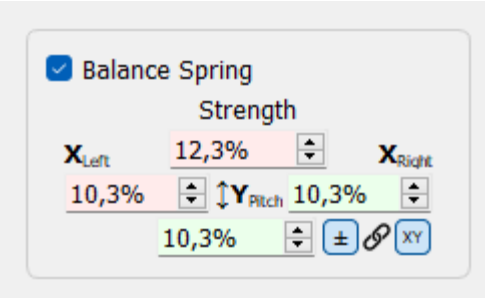
***Note:** Before adjusting the *"Balance Spring"* settings, it's crucial to disable Spring/Damper/Friction/Inertia effects. This ensures that adjustments made to the balance spring force settings are accurately reflected without interference from the other effects.

3.4.2. Adjusting "Balance Spring" Settings

Once the effects are disabled, you can proceed to adjust the "Balance Spring" settings. Follow these steps to fine-tune the spring force in different directions:

Within the VP Configurator, locate the effect box labeled ”Balance Spring”.

You'll find settings for adjusting the strength of the spring force in four directions: left, right, forward, and backward. Here's how to adjust each direction:



Left/Right: Increase or decrease the strength of the spring force to counteract imbalance caused by uneven weight distribution or grip extensions on the respective sides.

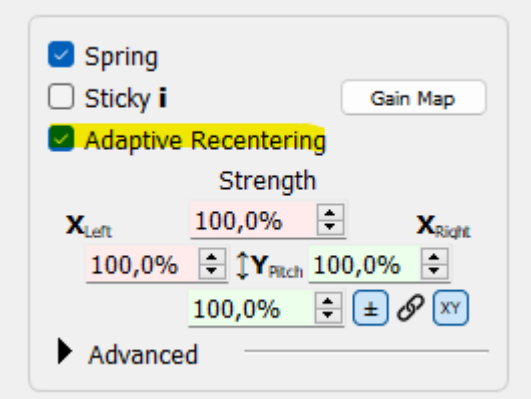
Forward: Adjust the spring force forward to counteract any tendency of the device to tilt forward, especially if the front end is heavier.

Backward: Similarly, adjust the spring force backward to counteract any tendency of the device to tilt backward, particularly if the rear end is heavier.

Experiment with different *Strength* settings and observe how the stick responds in various scenarios, such as holding and releasing it at different angles. Fine-tune the settings as necessary until you achieve the optimal values indicated by the stick staying in place after moving it to various angles.

3.4.2.1. Adaptive Recentering

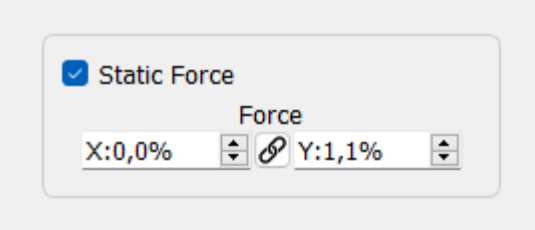
In addition to “Balance Spring” there is an additional facility to aid in reducing the error of the trimmed position - **Adaptive Recentering**.



Adaptive recentering aims to reduce the error between the trimmed position and the actual stick position by slowly adjusting the force to minimize the error. It works for any *Spring* class effects created by *directX* game or *telemFFB*.

The Adaptive recentering has limited authority based on spring strength, so on lower spring settings it might not have enough authority to bring the stick into position. It’s required to adjust the “Balance Spring” in that case.

3.4.2.2. Static Force



Static force setting is useful if the stick is very front or rear heavy. Increasing this setting is akin to adding a rubber band that adds a constant torque/force in a particular direction (can be positive or negative).

This effect is also very useful to simulate a counterbalance when building an FFB collective for example.

4. The VPforce TelemFFB Application

TelemFFB is a (work in progress) open source python based application which intakes telemetry from a simulator and uses that telemetry to produce various effects.

The latest version can always be downloaded from the repository release page here: https://github.com/walmis/VPforce-TelemFFB/releases

Supported simulators:

- DCS World
- Microsoft Flight Simulator
- IL-2 Sturmovik
- X-Plane 11/12

The repository is located on GitHub here: VPforce-TelemFFB

For DCS and IL-2, which support native FFB, the TelemFFB app is primarily leveraged to implement certain effects like gunfire, engine rumble and helicopter ETL shaking (among many others). However, there are some additional 'FFB type' effects which are implemented such as deceleration force and g-loading effect.

For MSFS and X-Plane, which ***do not*** have native FFB support, TelemFFB also implements dynamic axis FFB in addition to most of the effects previously mentioned for DCS.

The currently active branch in the repository is the WIP branch and the remainder of this section covers the features and functionality held within that branch.

If you are new to TelemFFB and want to see how far it has come, or you are an old-timer stuck in your ways, or if you just hate puppies, dislike shiny things and enjoy pain, you can check out the legacy v1.0.0 release. The documentation for the legacy version is retained in Appendix B.

Important differences from Legacy Version

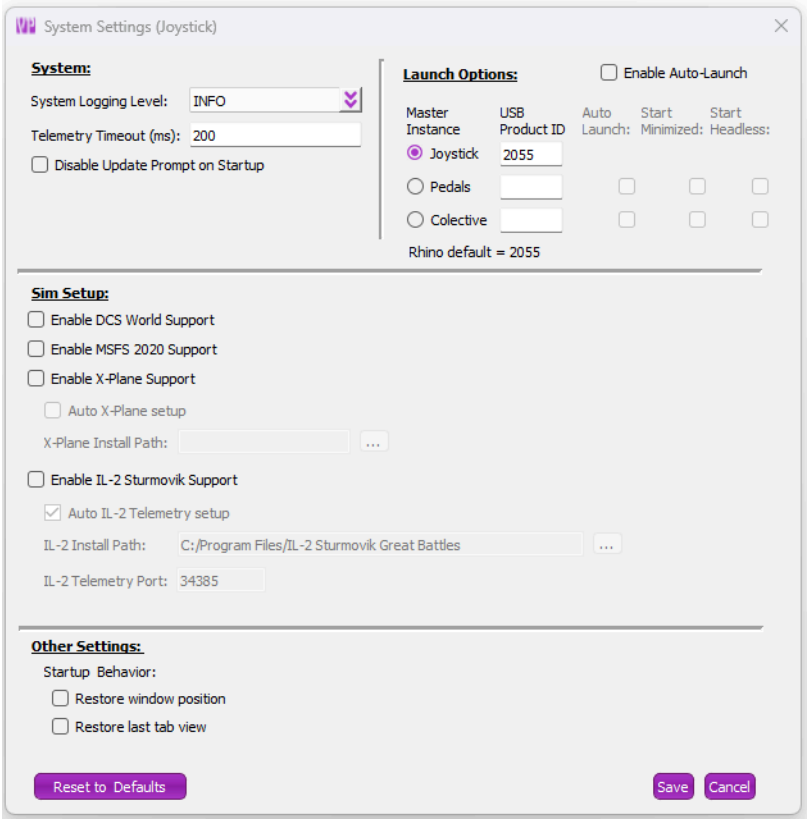
Previous versions of TelemFFB used a flat '.ini' style configuration model. Each setting was manually configured with an enable/disable flag and associated strength value.

- Versions of TelemFFB from the development branch after January, 2024 make use of a UI based configuration model which abstracts the complexity of the configuration from the user. Enabling/Disabling and adjusting effects is a simple matter of interacting with the UI.
    - If you were previously using a 'user config override file', Telem FFB will detect this and you will be prompted on whether you want to convert your previous settings to the new settings model. See the Migration section for details.
- TelemFFB system settings are now stored in the system registry at the following key location: **`Computer\HKEY_CURRENT_USER\Software\VPforce\TelemFFB`**
- The user configuration file and log files will be auto-generated and are located in: **`%LOCALAPPDATA%\VPForce-TelemFFB`**
    - It is no longer necessary to use the '-c' or '-o' options to specify an override file. The user override file is automatically managed by TelemFFB.
    - You can access the config/log folder from the **System** menu in TelemFFB
- While the legacy '-D' and '-t' runtime flags are still supported, they are generally no longer required. You can use the options to start an instance of TelemFFB for a specific physical device and/or device type, however it will be far more common to use the auto-launch setup if you have multiple FFB devices.

4.1. First-Time Setup

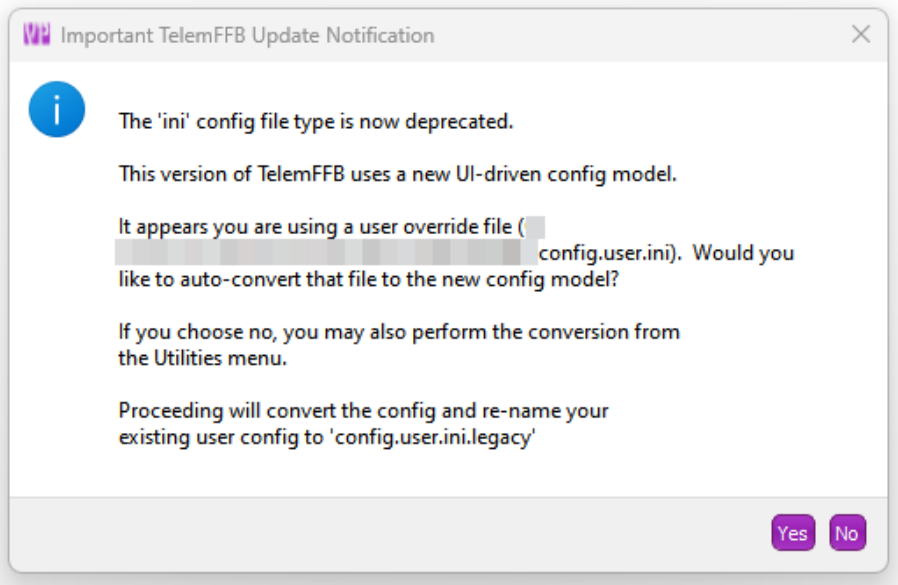
4.1.1. New Installations:

The very first time you install and launch TelemFFB, you will be greeted by the system settings window. Follow the guidelines in the systems setting section for setting up TelemFFB.

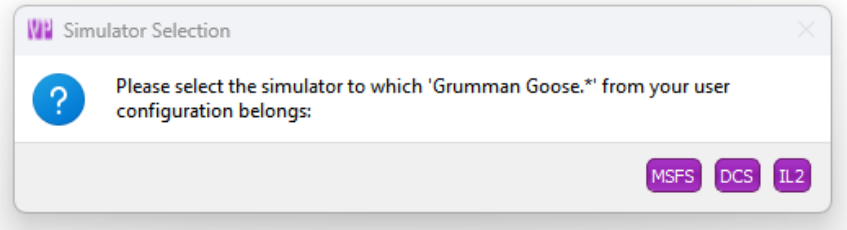


4.1.2. Migration from Legacy Version 1.0

If you were previously using a user config override file, whether it be the default “config.user.ini" or if you were passing another file using the ‘-o’ runtime flag, TelemFFB will detect this and offer to convert your previous settings into the new settings framework. If you have multiple devices, you will have to perform this on each one. It all gets saved into the single userconfig.xml file. If there are existing settings saved already, this will append to them. If an existing setting is for the same sim, setting name, and device then the existing value will be overwritten.



During the conversion, you will be prompted to identify the applicable simulator for any non-default aircraft that the conversion script finds.



After all of your configuration has been converted, you will see the following dialog.

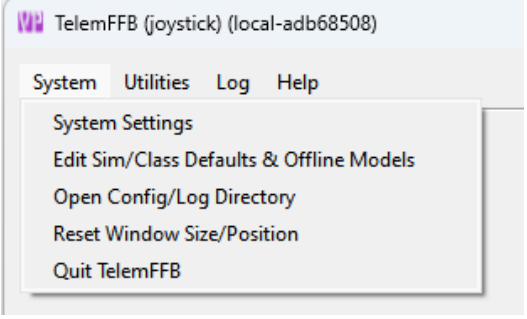
Following the conversion process, if this is the first time you are starting the new version of TelemFFB, you will be greeted by the system settings dialog for initial setup. Follow the guidelines in the systems setting section for setting up TelemFFB.

4.2. General Application Overview

TelemFFB is laid out with a menu bar, a simulation status area and the view tabs at the bottom. Refer to the sections below for details on each.

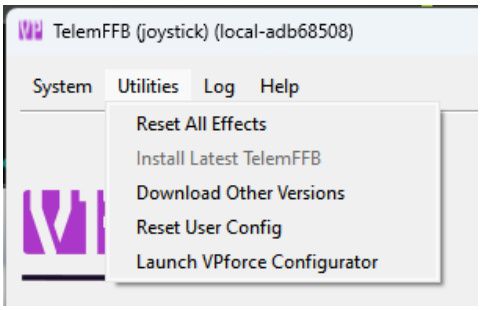
4.3. The Menus

4.3.1. System Menu



- **System Settings** - covered here
- **Edit Sim/Class Defaults & Offline Models** - While the Settings tab of the main window allows you to edit the current aircraft, this settings editor allows you to see model settings for any aircraft, without the sim needing to be active, and also enables changing default settings that apply to an entire sim or class of aircraft. More detail is in a section below.
- **Open Config/Log directory** - opens the folder in your user local appdata where logs and settings are stored
- **Reset Window Size/Position** - resets to default
- **Quit TelemFFB** - closes the application and stops all effects.

4.3.2. Utilities Menu

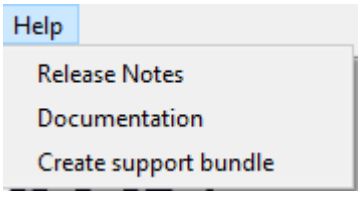


- **Reset all Effects** - Reset the VPforce device and clean up any lingering effects. **Note: Is destructive to any active effects being generated by a simulator.**
- **Install Latest TelemFFB** - Start the auto-update process. Only active if an update is available and the update prompt is disabled or was dismissed on startup.
- **Download Other Versions** - opens a webpage where you can select legacy versions to download.
- **Reset User Config** - Removes all user configured settings from TelemFFB and reverts to 'factory defaults' for all effects settings. Note that when this is executed, a date-time stamped backup of the existing user configuration is saved in the TelemFFB folder in AppData/Local
- **Launch VPForce Configurator** - Cross launches the VPforce configurator app to set up your device

4.3.3. Log Menu

- Open Console Log - Open the log window

4.3.4. Help Menu

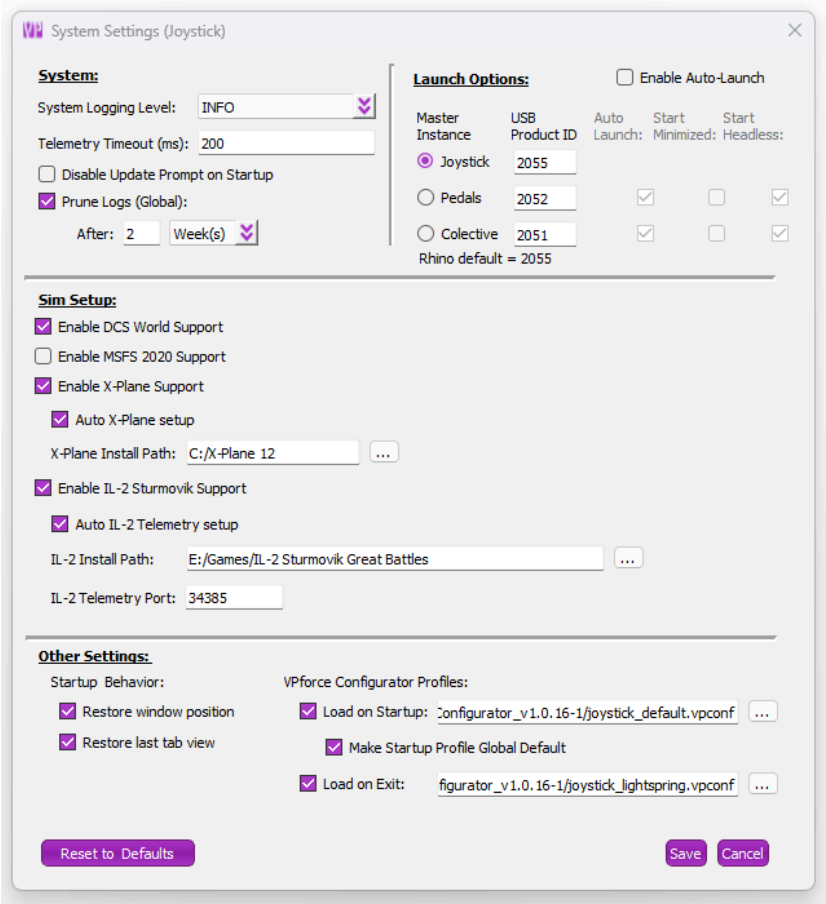


- **Release Notes** - shows any release notes for this version of TelemFFB

- **Documentation** - opens this manual
- **Create Support Bundle** - Opens a file dialog and creates a zip file containing your TelemFFB system settings, any user settings you have stored, and your Log folder..

4.4. System Settings

In the System Menu, choose System Settings:



4.4.1. System Options

These settings are unique per device instance of TelemFFB

- **System Logging Level**
  - Control the logging level for an instance of TelemFFB
- **Telemetry Timeout**
  - Control the telemetry timeout value for an instance of TelemFFB
- **Update Prompt Control**
  - Enable/Disable the new-update prompt for an instance of TelemFFB when starting up.
- **Prune Logs**

- Enable log pruning. Archived log zip files that are older than the configured threshold will be automatically deleted upon TelemFFB startup.

4.4.2. Launch Options

These settings are global for any instance of TelemFFB and affect how the application starts up and communicates with one or more FFB devices.

- **Enable Auto-Launch**
  - Tick this checkbox to enable the auto-launch feature which will start multiple instances of TelemFFB to communicate with multiple FFB devices. See the section on running with multiple FFB devices for details.
- **Master Instance Radio Buttons**
  - Independently of the auto-launch feature, the selected radio button defines the device that TelemFFB will connect to when it is launched.
  - When combined with the auto-launch feature, the selected device will act as the master instance for any additional spawned instances of TelemFFB.
- **USB Product ID**
  - Enter the USB Product ID that is configured for a given device (as configured in VPforce FFB Configurator)
- **Instance Auto Launch Options**
  - Auto Launch
    - Enable or disable auto-launching of an instance when the master instance loads.
  - Start Minimized
    - Start the selected instance with its window minimized
  - Start Headless
    - Start the selected instance with its window hidden (can be revealed from the master instance **window** menu)

4.4.3. Startup Options

- **Start with Windows**
  - When enabled, an entry will be added to the Windows registry that will start TelemFFB automatically when Windows starts
  - **Note:** Only available with the EXE distribution of TelemFFB. This option will be disabled when running from source
- **Start in System Tray**
  - When enabled, TelemFFB will start up minimized to the system tray. The main window can be recalled by double-clicking the system tray icon or from the right-click context menu on the system tray icon.
  - Note: This is mutually exclusive with the Start Minimized option. Only one or the other may be enabled
- **Start Minimized**
  - When enabled, TelemFFB will start with its main window visible, but minimized to the taskbar.
  - Note: This is mutually exclusive with the Start in System Tray option. Only one or the other may be enabled
- **Closing App Sends to Tray**
  - When enabled, pressing the window close button will simply minimize the application to the system tray.
  - You can fully exit TelemFFB from the System menu or from the right-click context menu on the system tray icon.

4.4.4. Sim Setup Options

These settings are global for any instance of TelemFFB.

To enable a given simulator, simply tick the checkbox. The applicable telemetry engine will start automatically when the settings are saved. There is no need to restart TelemFFB.

For X-Plane and IL-2, TelemFFB needs to be told where the installation folder is so that it can perform the necessary plugin install (x-plane) or telemetry setup (IL-2). For X-Plane this is also important so that the plugin will stay up to date as needed if the plugin is updated between versions

4.4.5. Other Settings

These settings are unique per instance of TelemFFB

- **Restore Window Position**
  - When enabled, TelemFFB will remember where the window was positioned the last time it was run and restore the window to that same position
- **Restore Last Tab View**
  - When enabled, TelemFFB will remember the window size for each tab the last time it was run. It will also restore these sizes and remember the last tab that was viewed the last time it was run.
- **VPForce Configurator Profiles**
  - Define a profile to load on TelemFFB startup and/or exit
  - See the section on Dynamic Configurator Profiles for more details

4.5. Application Main Window

4.5.1. Sim Status Area

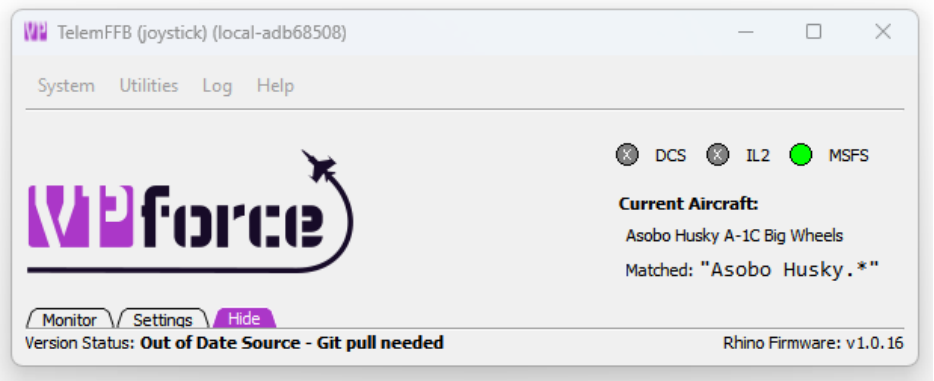
The colored icons show the status of receiving telemetry from those sims:

Sim Disabled In System Settings	Sim Enabled, Telemetry not yet received
Receiving Telemetry from Sim	Receiving Telemetry, Sim Paused
Error condition exists - Error message will replace Aircraft Status Area	

The “**Current Aircraft**” will display the name of the aircraft as received in the telemetry and the profile “match string” that was used to load the applicable settings.

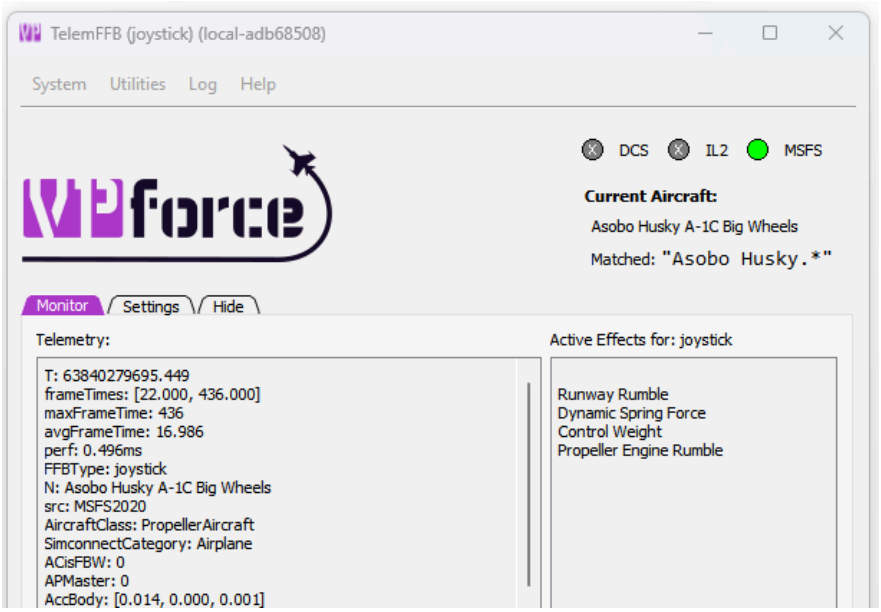
4.5.2. Hide Tab

The Hide tab is the simplest and reduces information shown to the bare minimum:



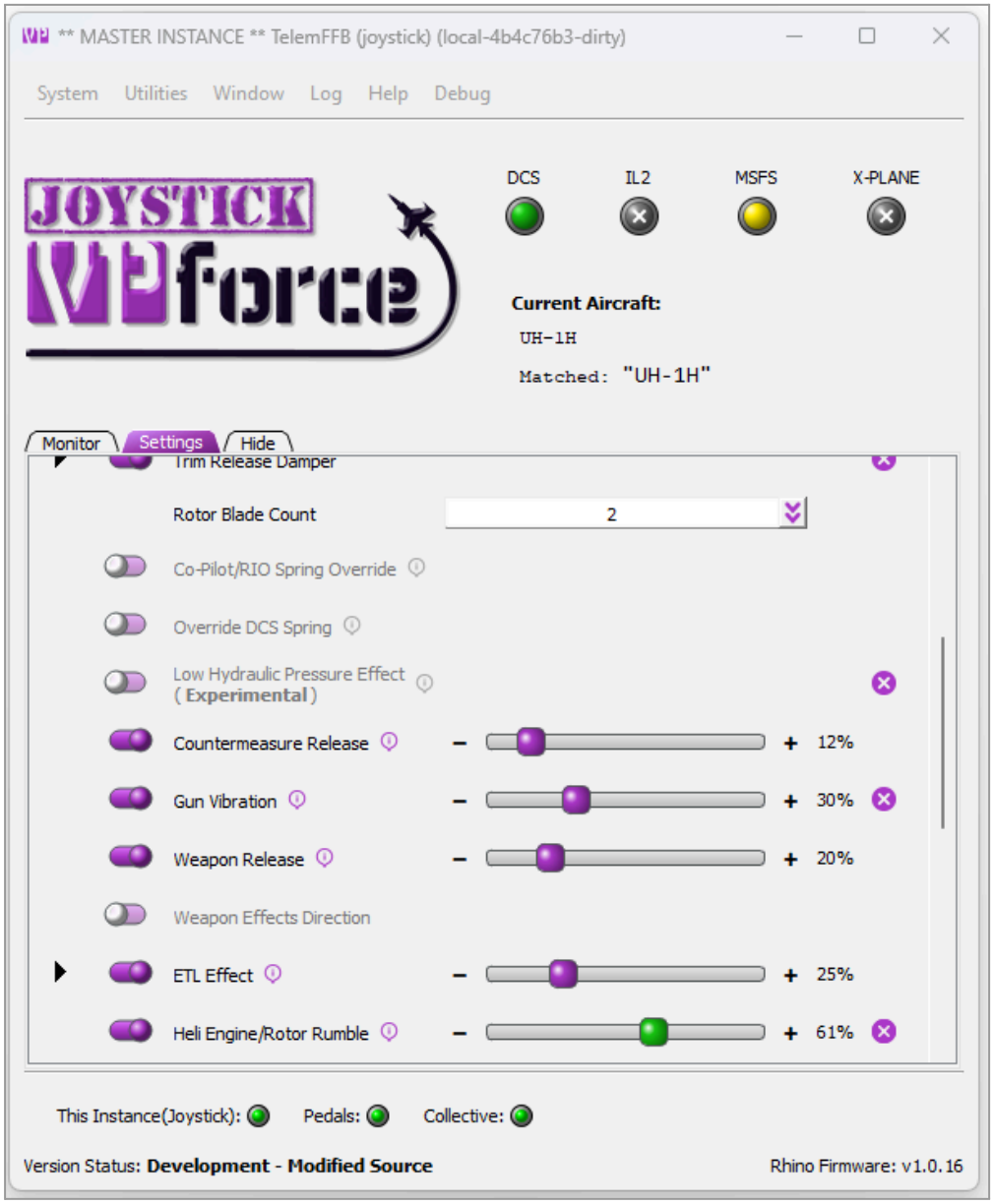
4.5.3. Monitor Tab

The Monitor tab shows received telemetry data and effects that are currently active:



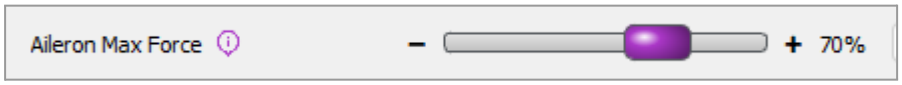
4.5.4. Settings Tab

The Settings tab allows you to edit all possible forces and effects for the current aircraft loaded in the simulator. This section describes the interface, details about each setting are in other parts of the manual. Changing any setting has an immediate effect. To edit the global settings for a simulator, aircraft category or an aircraft which is not currently loaded in the simulator, you can use the Sim/Class & Offline editor



Sliders with no toggles on the left are always active. Change the setting by **dragging the handle** to a new position or by pressing the **-/+ buttons**. You can also make adjustments while hovering over the slider by **holding shift and using the mouse wheel** or by using the **left/right** or **up/down** arrow keys.

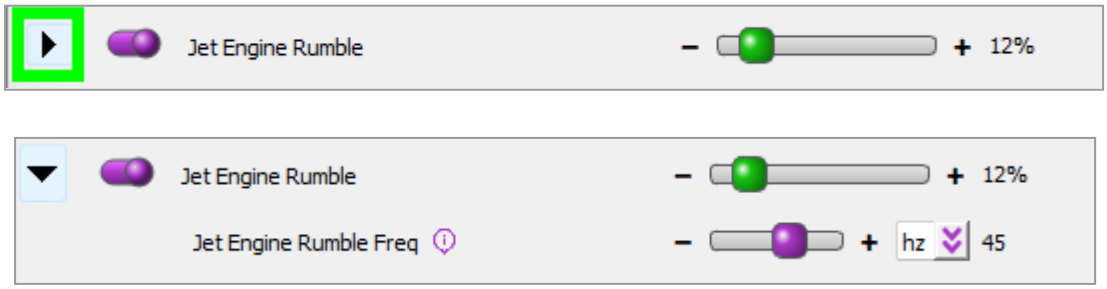
Example:



Effect sliders have a toggle to enable or disable that effect. You can quickly toggle on/off an effect. When the setting is off, your intensity setting is retained. **The handle will also turn green when that effect is active**.

Example

Some settings have additional parameters. You will see an expander button next to them. **Click the expander** to see additional settings, and again to collapse:



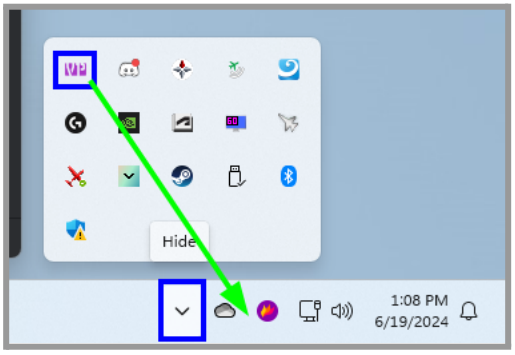
Any setting you have modified will show a ‘x’ icon on the right side. You can **click this icon** to return the setting to the default:

For settings where a unit is used, there is a dropdown of acceptable units:

Some settings require a grip button assignment before use. **Click the button** on the screen and then press the desired grip button before the timer expires:

4.6. System Tray

When TelemFFB starts, a system tray icon with a context menu is added to the windows taskbar. By default, the icon will be accessible from the expander button in the system tray. You may choose to drag the icon into the pinned icon areas of your task bar so that it is always visible.



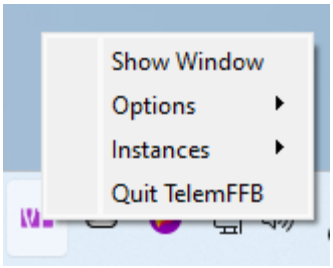
You can force the TelemFFB window to show by double-clicking the icon or from the context menu.

When you start a simulator, the system tray icon will change colors to indicate the current status of the sim, very similar to the status indicator icons in the main TelemFFB window.

- Sim is running, receiving telemetry
- Sim is paused
- Error condition. There will also be a system tray notification with the error information and the error message will be visible on the TelemFFB main window.

4.6.1. System Tray Context Menu

Right-clicking on the system tray icon will open the context window. There are several items.



- **Show Window**
  - Forces the TelemFFB window to show itself if it is hidden and also come to the front if it is minimized or hidden by other windows.
- **Options**
  - Provides options for toggling the Start to / Close to tray options. These toggles are identical to the checkboxes in the system settings page. If you change the options here, they will take effect immediately and you will see the change reflected in the system settings.
- **Instances**
  - If you are running with multiple VPforce devices, this menu will have options to explicitly show the window for each of the additional instances of TelemFFB
- **Quit TelemFFB**
  - Exits the application

4.7. Adding New Aircraft Support

While TelemFFB has default profiles for many aircraft already, there are many hundreds of possible aircraft between all of the simulators that are supported. So, inevitably you will run into an aircraft that does not already have a built-in profile

There are two ways to add support for new or unknown aircraft to TelemFFB.

- Dynamically when a new aircraft is loaded in a simulator
- Via the Offline Settings manager, independent of any running sim

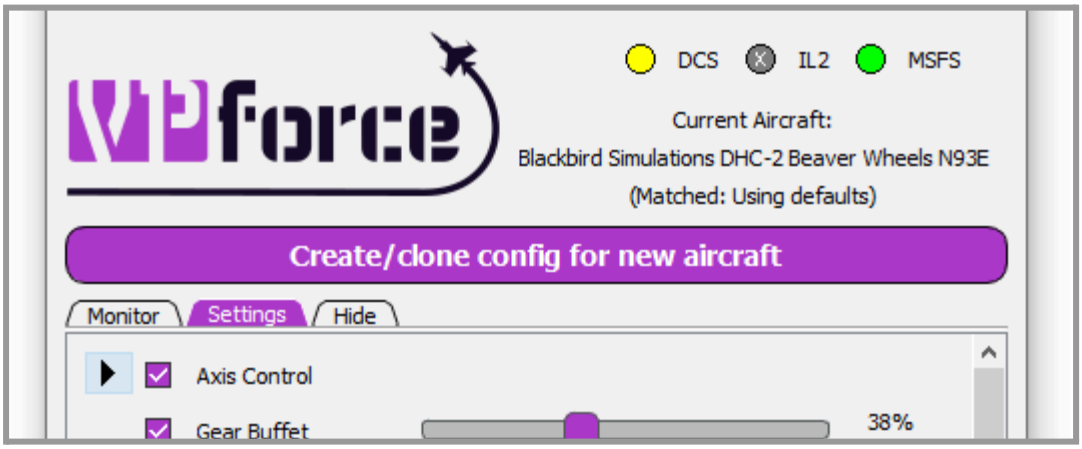
4.7.1. Accessing the New Aircraft Dialog

4.7.1.1. Dynamic from Main Window (Recommended)

If you are actively using a simulator and you load into an aircraft that does not have a default settings profile, you will see a reduced set of effects and large “clone new aircraft” button will appear

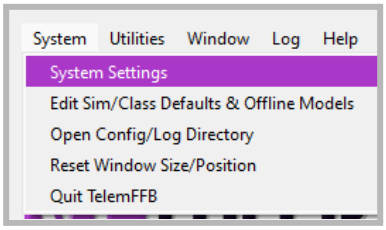
This method is preferred over the offline editor since TelemFFB will already know the proper name string for the aircraft.

- Simply click on the “Create/clone config for new aircraft” button

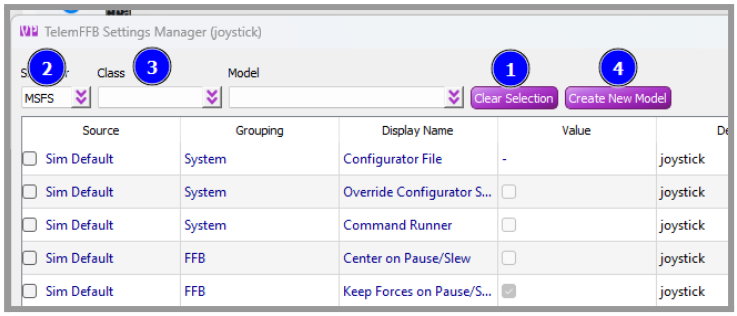


4.7.1.2. Offline Editor

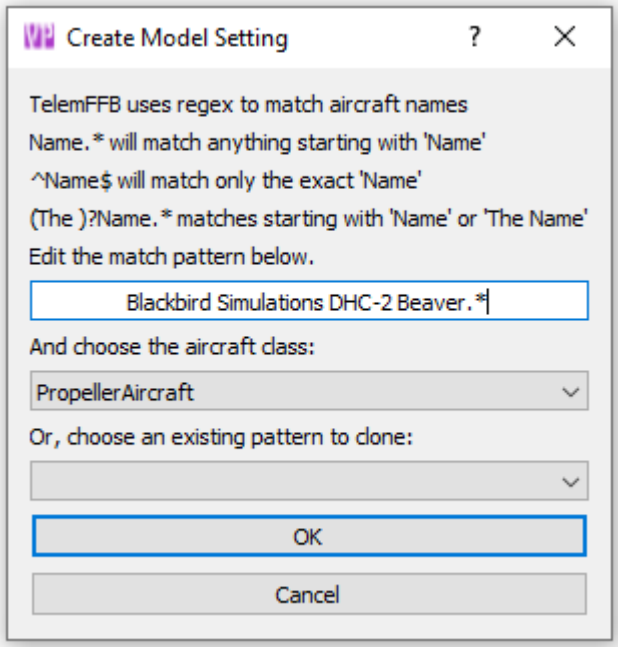
- Open the Offline Manager



- In the resulting Settings Manager window:
    - First Clear the selection (1)
    - Select the Simulator to which you want to add an aircraft (2)
    - Select the Aircraft Class which matches the aircraft you are adding (3)
    - Select the Create New Model Button (4)



4.7.2. Using the New Aircraft Dialog



After accessing the dialog via one of the two methods above:

- If the dialog was opened via the dynamic loading of an unknown aircraft, the “match pattern” will be pre-filled with the string that was received by TelemFFB
    - Edit the name into a match pattern so that other liveries or variants of the same aircraft use these settings.
    - Note: Some suggested match string wildcard combinations are also provided in the pulldown.
- If the dialog was opened from the Offline editor, you will need to manually enter the match string for the aircraft.
    - There is no universal way to find out what the string will be for a given aircraft from a given sim.
    - For MSFS it is the ‘name” in the aircraft.cfg file. For DCS it is the “name” field in the given aircrafts <aircraftname>.lua file.
    - For this reason it is recommended to use the dynamic method since TelemFFB will already have the correct name to base the match string off of.

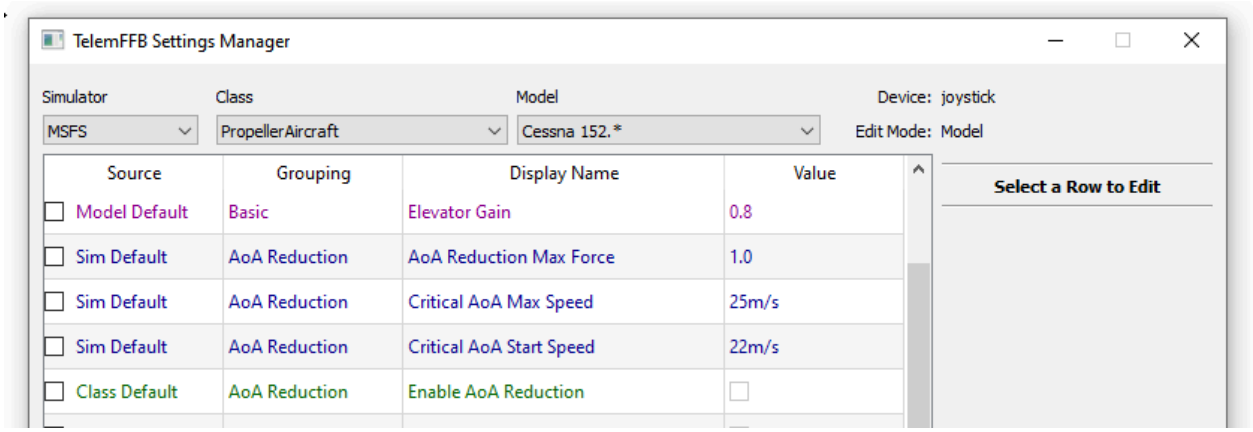
- Select the type of aircraft it is from the aircraft class pulldown
- Optionally, select an existing aircraft that you want to clone settings from. The newly formed aircraft entry will duplicate all of the current settings from the cloned aircraft
- Click OK

4.8. Offline/Global Sim/Class Configuration

While the main screen settings sliders are used for the currently loaded aircraft, you can change any of the default settings to apply to anything in that sim, any aircraft of that class, or previously stored individual model settings to your liking. From the **System** Menu, choose **Edt Sim/Class Defaults & Offline Models**.

TelemFFB settings are applied in a hierarchy: Simulator -> Aircraft Class -> Aircraft Model. A setting in a higher level will apply to the aircraft unless a lower level setting exists. Setting hierarchy is shown in the Source column, and by color- Sim settings are blue, Class settings are green and Model settings are magenta. The Source column shows where the current setting was read from in the hierarchy. It also shows whether it came from a default setting, or was changed by the user.

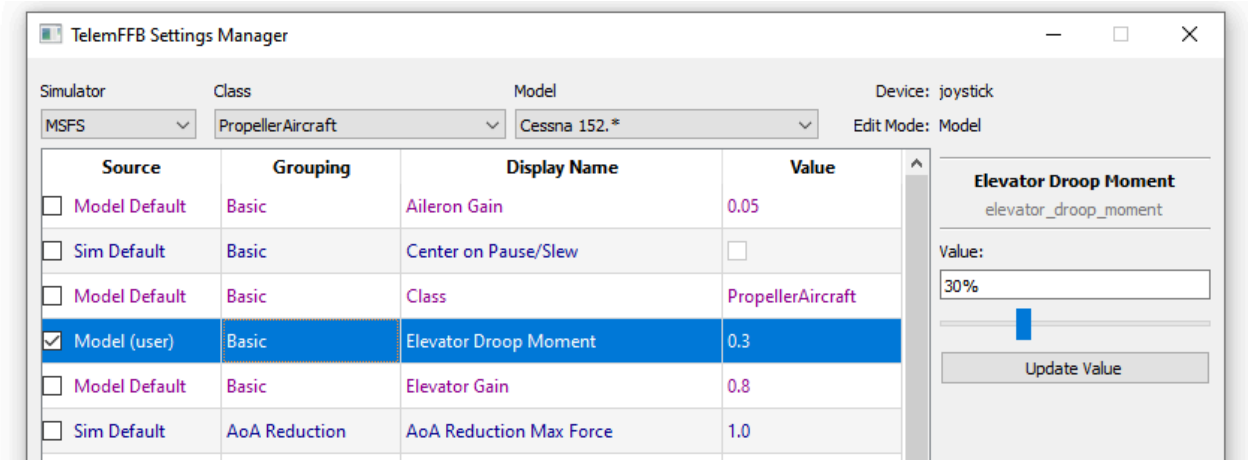
Interface



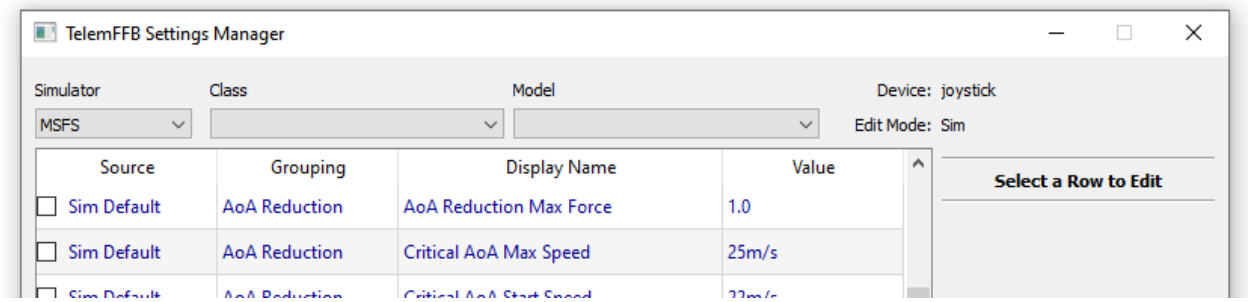
Across the top row are the dropdowns for Simulator, Class, and Model. You can change the dropdowns to edit other Simulator, Class or Model settings. In the property manager on the right side, you can drag the slider or type a new value in the box and click Update Value. The setting will apply immediately. The property manager interface will be slightly different for other datatypes such as a checkbox for enable/disable or dropdowns for valid values. Many options have some information to explain their usage.

Note the **Edit Mode**, shown in the upper right of the window. These modes allow you to edit the settings at that hierarchy level. You can change the Edit Mode by selecting another, or blanking out, the dropdown menus. When you change a setting in Sim mode, it will apply to all aircraft in that simulator. In Class mode, such as 'Helicopter', you are changing settings for all helicopters. In Model mode, you can override something that was set for Sim or Class, and is the mode that the main window works in.

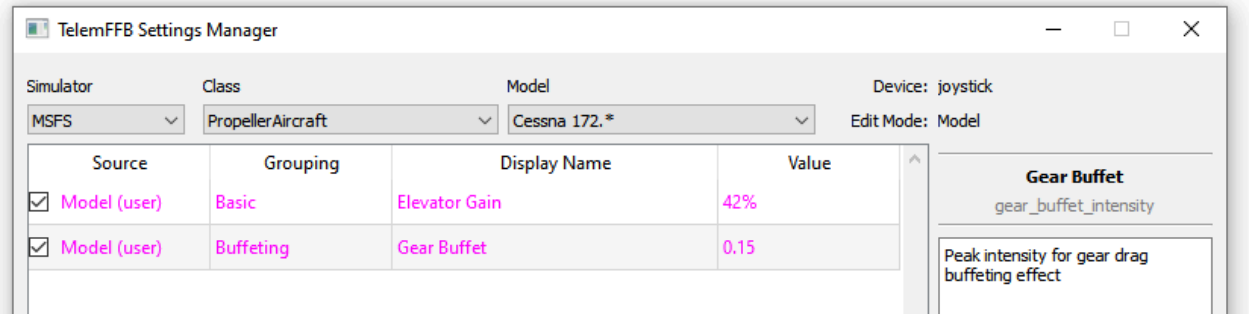
To change a default setting, it first must become a *user* setting (vs. a default setting). If you click on rows of default settings, you'll find that it has no effect. Click on the checkbox to turn any default setting into a user setting in that mode, and then it will be selectable.



Selecting a blank entry in class or model will show the defaults for the next higher category. Here, both class and model are blank, so we are editing **Sim** defaults:



The Show Defaults toggle on the bottom left will show/hide settings that are not user customized.



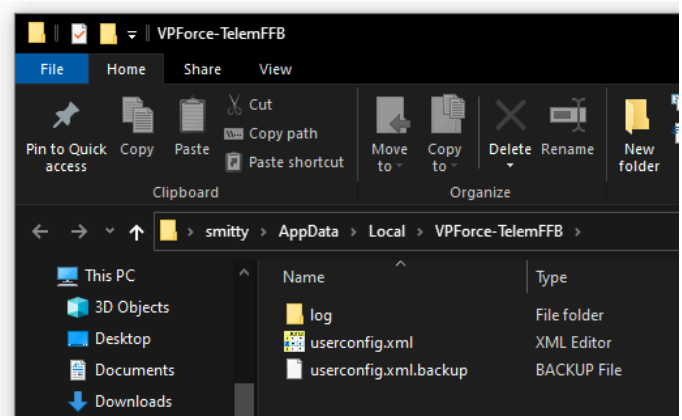
Also in the bottom left corner is a **Revert** button, which will restore any changed settings back to the moment that the Settings Window was opened. On the bottom row is a direct link to this section of this manual.

4.9. The User Config File

All user settings for all simulators (and all devices, if applicable) are stored in:

`C:\Users\{username}\Appdata\Local\VPForce-TelemFFB\userconfig.xml`

You can select the menu **System->Open Config/Log Folder** in the main window to quickly open the folder. The xml file will be helpful to troubleshoot any issues. If you are getting support, it may be useful instead to create a support bundle using the option in the Utilities menu, and send that instead, as it will contain a complete package of your settings.



4.10. Running TelemFFB with multiple VPforce FFB devices

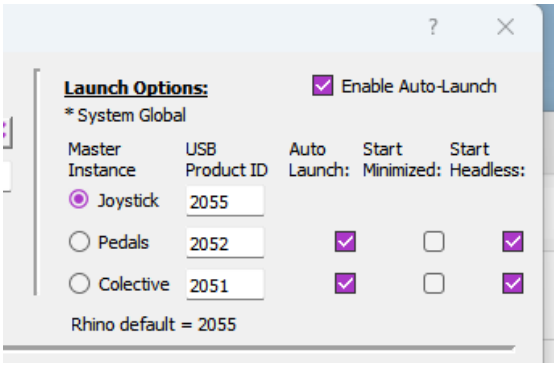
With the availability of the VPforce DIY kits, people are developing their own FFB devices such as rudder pedals, and even collectives. While DCS does not support FFB on the rudder or collective axes (nor MSFS or IL2), it is still possible to play all of the effects that TelemFFB offers through any VPforce device.

By default, TelemFFB attempts to connect to the VID:PID address that is specific to the Rhino Joystick Base. The VID for all VPforce control boards is ‘FFFF’. The default PID for the Rhino Joystick Base is ‘2055’. The PID can be viewed (and modified) in the VPforce FFB Configurator utility:

With previous versions of TelemFFB, it was necessary to start multiple instances of TelemFFB using ‘-D’ to specify the VID:PID and ‘-t’ to specify the device type (joystick/collective/pedals) and each configuration had to be maintained separately.

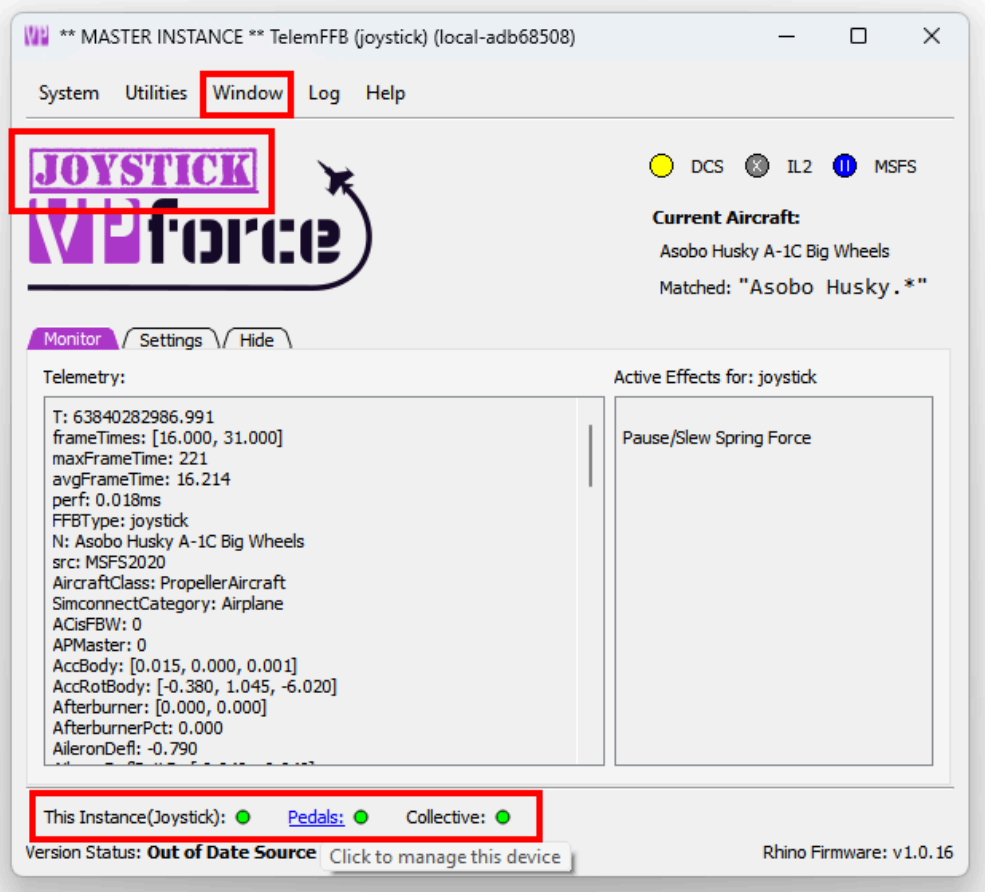
The whole process of running with multiple devices has been largely automated.

In the system settings Launch Options section, you can configure the PID values for each of your FFB devices.

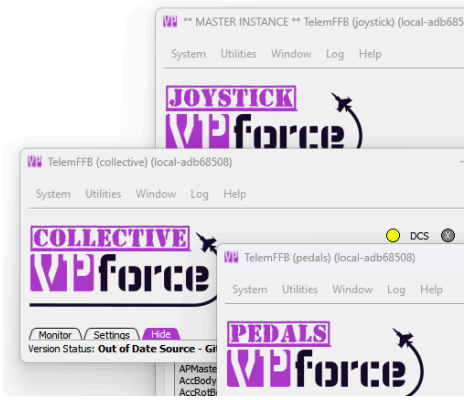


You can also choose which additional devices you would like to automatically launch when the master instance of TelemFFB is started. These additional child instances can be started in normal, minimized or headless modes.

After starting TelemFFB using the auto-launch mode, you will see a slightly different interface.



The first thing you will notice is the large device label. If you chose to launch the child instances with their windows shown, this is an easy indicator as to which instance you are currently looking at.

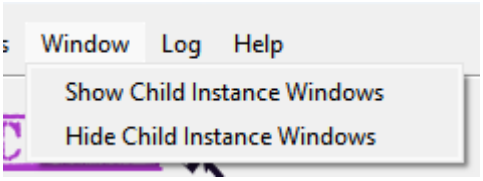


From the master instance, you can also use this label as a button to switch between the different child instance settings personas. Clicking on the label will switch the master instance of TelemFFB between devices and allow you to adjust settings for that device. It will also update the active effects window to reflect the currently playing effects for that device.

The second difference is the instance status icons at the bottom of the window. These serve two purposes. The color of the dot (green or red) indicates the status of the inter-process communications between the master and child instances. If the icon turns red, that is an indication that there is a problem with the communication and that instance may have crashed.

You can also click on the device name to switch configuration modes for the master instance.

Lastly, there is an additional 'Window' menu that will allow you to show and/or hide the child instance windows.



4.11. Dynamic VPforce Configurator Profile Assignment

Often it is desirable to have different VPforce Configurator settings in place for different aircraft, types of aircraft or simulators.

While this can be accomplished manually by simply loading a profile in configurator and then applying it, that is a tedious process.

Fortunately, this process can be automated using TelemFFB. There are a variety of ways that VPforce Configurator profiles (hereafter called vpconf/configurator profiles) can be automatically loaded onto your device via TelemFFB.

****Note** -** It is important to understand the hierarchical effect of defining configurator profiles and various levels of specificity.

The startup/exit profiles will *always* load on startup/exit.

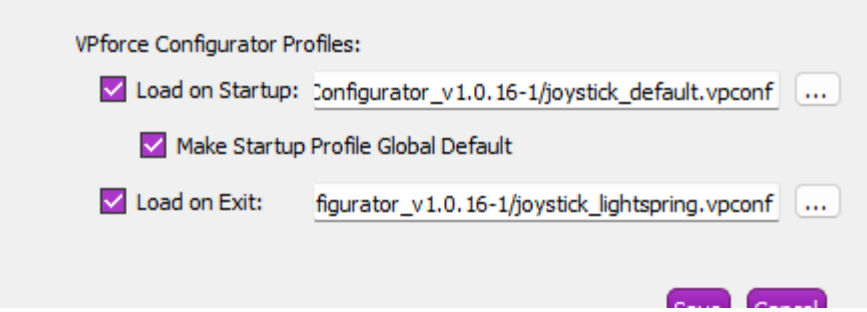
The “global default” will always load if a prior aircraft/class/sim loaded a specific profile and a newly loaded aircraft does not have any other more specific definition.

The more specific the definition, the higher the precedence. A configurator profile defined for a specific aircraft will supersede all other profiles when that aircraft is loaded.

Similarly, “aircraft class” (helicopter/prop/jet, etc) will supersede “sim” (DCS, MSFS, etc).

4.11.1. Startup/Exit configurator profiles

In the system settings options there are fields where you can select discrete profiles that TelemFFB will always push to the device when it starts, or when it exits. This can be useful for keeping a low force profile on your device while not in use, but loading a profile with higher effect settings gains when you start TelemFFB.



4.11.2. Global Default configurator profile

Also in the system settings, you can set the TelemFFB startup profile to be used as a global default for any aircraft or sim. This is useful if you only have one or two specific aircraft that you wish to apply custom profiles to.

When an aircraft with a custom profile is loaded, TelemFFB will push the profile defined for that aircraft to the device.

At some point later, if another aircraft is loaded and it does not have a profile defined at either the Aircraft, Class or Simulator level, TelemFFB will automatically revert the configuration on the device to the global default value.

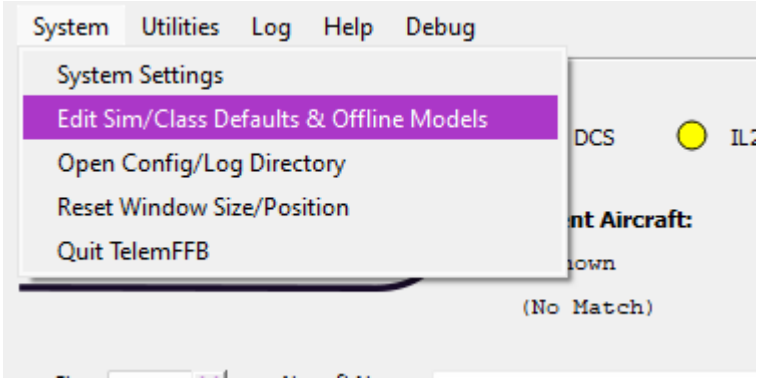
4.11.3. Assigning a configurator profile to a specific aircraft (real time config)

When loaded into an aircraft, simply select the Configurator File selection button in the TelemFFB settings tab and navigate to and choose the configurator profile. This will be stored in your user configuration and every time that aircraft is loaded, TelemFFB will push the profile to your device.

4.11.4. Assigning a configurator profile to an aircraft/class/sim (Offline Editor)

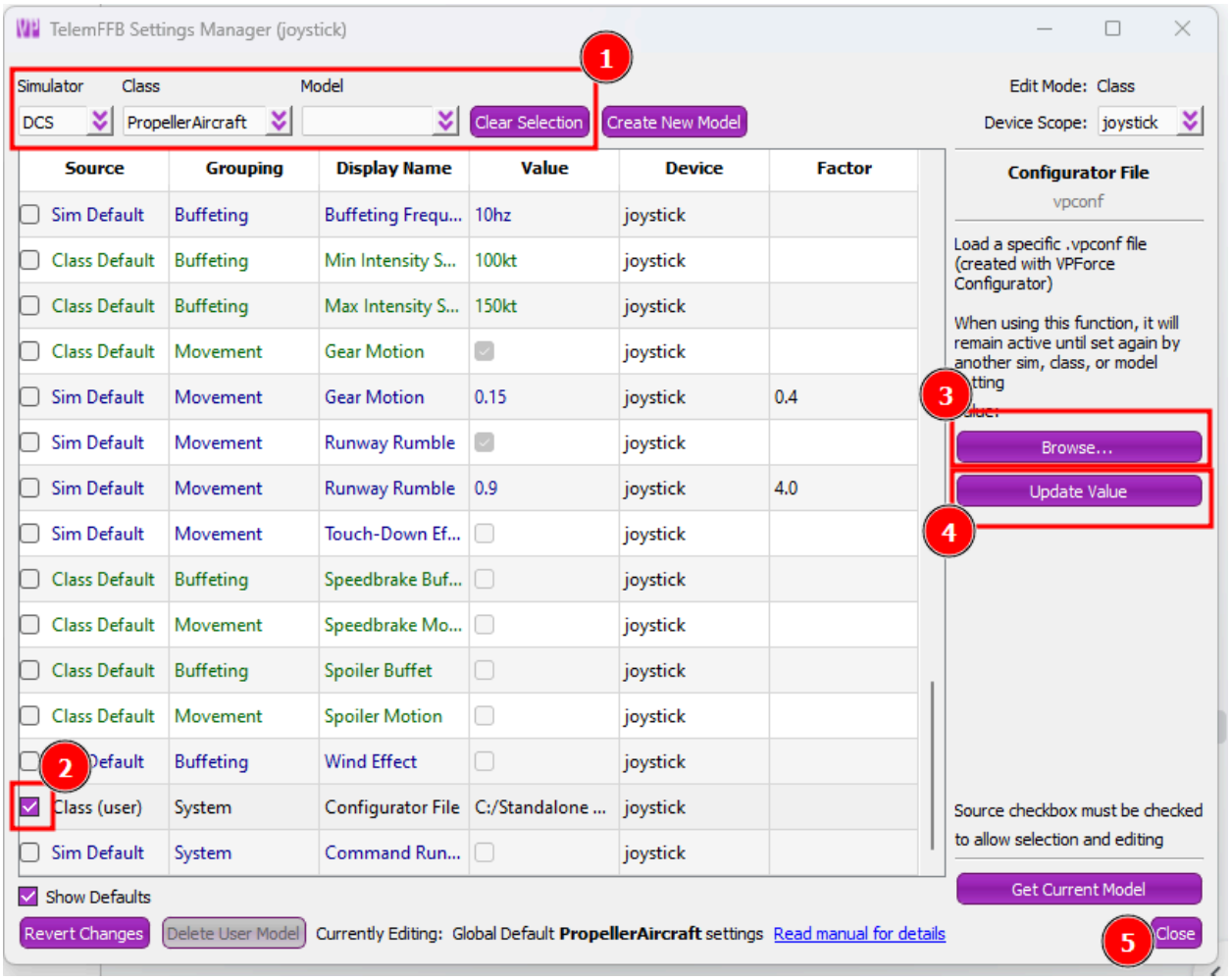
You can also assign a profile to an aircraft that is not actively loaded in TelemFFB. In this same way, you can assign a profile that will be loaded for all aircraft of a given class or even any aircraft for an entire sim.

In the System->Edit Sim/Class Defaults & Offline Models menu:



1. Use the 'Clear Selection' button and the pull-down boxes to select the Sim, Class or aircraft you wish to modify.
2. Scroll down and find the Configurator File line item, enable the checkbox if it is not already enabled. If it is enabled, click on the row.
3. In the right hand pane, select the browse button and select your configurator profile

4. After selecting the profile, you *must* press the Update Value button for the change to take effect.
5. When finished, close the editor.



4.12. Dynamic Configurator Gain Overrides

In addition to (or in lieu of) pushing a whole VPforce Configurator (“vpconf”) profile to the device, you can also configure dynamic adjustments to the individual effect type gains as if you were adjusting them directly from the configurator app.

As with any other setting in TelemFFB, this can be done at the Sim, Aircraft Class or specific aircraft level.

For Sim/Aircraft Class, these must be accessed from the Offline/Global Sim/Class Editor.

For individual aircraft, it is easiest done straight from the settings page on the main window.

4.12.1. How it works

First, it is important to understand how this works.

- There is always a baseline set of gains stored in TelemFFB while running:
  - The current gain values on the device are captured when TelemFFB is started.
  - Any time a VPforce Configurator (“vpconf”) profile is pushed to the device by TelemFFB, the new gains on the device are read and remembered by TelemFFB for later use.
    - This includes VPforce Configurator profiles that are pushed as the “TelemFFB startup Profile”, and any that are pushed as part of a Sim/Class/Aircraft configuration
- If an aircraft that has a Configurator Gain Override configured is loaded
  - The gain values will be set ***after*** any “vpconf” profile is pushed. This means that the gain overrides will supersede the “vpconf” profile gain settings.
- Subsequently, if an aircraft that ***does not have*** a Configurator Gain Override configured is loaded into the sim
  - TelemFFB will revert the gain settings on the device to the last baseline value. This will either be the gains that were read at startup, or the gains that were read after the last time a configurator profile was loaded.

- When exiting, TelemFFB will re-push the gain values that were initially read on startup so as to leave the device in the same condition it was found.

    simplest example:

4.12.1.1. Example Configurations

Several examples follow in an attempt to describe the behavior of the Configurator Gain Overrides and its interaction with the Dynamic VPForce Configurator Profile feature. Each example walks through the behavior from startup of TelemFFB, through 2 different aircrafts loading with different settings and finally exiting TelemFFB.

Example 1

Simplified using a single effect type in the example. Synopsis below:

- ❖ TelemFFB not running
- ❖ Current Spring Gain on device - %50
- ❖ No “startup vpconf” configured.
- ❖ No “vpconf” specified for example loaded aircraft
- ❖ First example aircraft has spring gain override configured at %100

- Start TelemFFB
  - Spring effect gain read at %50 and value stored for later use
- Load aircraft with override configured with spring at %100
  - Spring gain of %100 gets set to device
- Load aircraft with *no* override configured
  - TelemFFB pushes original %50 that was read on startup
- Exit TelemFFB
  - TelemFFB pushes original %50 that was read on startup as final measure to ensure same state as startup.

Example 2

Simplified using a single effect type. Synopsis below

- ❖ TelemFFB not running
- ❖ Current Spring Gain on device - %50
- ❖ “startup vpconf” configured with spring gain = %75
- ❖ No “vpconf” specified for example loaded aircraft
- ❖ First example aircraft has spring gain override configured at %100

- Start TelemFFB
  - Spring effect gain read at %50 and value stored for later use
- Startup VPconf Profile pushed
  - Spring effect gain read at %75 and value stored for later use
- Load aircraft with override configured with spring at %100
  - Spring gain of %100 gets set to device
- Load aircraft with *no* override configured
  - TelemFFB pushes spring gain %75 that was read after the startup vpconf was pushed
- Exit TelemFFB
  - TelemFFB pushes original %50 that was read on startup as final measure to ensure same state as startup.

Example 3

Simplified using a single effect type. Synopsis below

- ❖ TelemFFB not running
- ❖ Current Spring Gain on device - %50
- ❖ “startup vpconf” configured with spring gain at %75
- ❖ First example aircraft has “vpconf” configured with spring gain %80 ***and*** a spring gain override set at %40

- Start TelemFFB
  - Spring effect gain read at %50 and value stored for later use
- Startup VPconf Profile pushed
  - Spring effect gain read at %75 and value stored for later use
- Load aircraft with vpconf set at %80 and spring gain override configured at %40
  - TelemFFB pushes new vpconf
    - new gains are read and stored for later use
  - TelemFFB pushes gains from the override config
  - The net result is that the gains on the device will be whatever is in the override config since it happens last
- Load aircraft with *no* gain override and *no* vpconf configured
  - The following behavior depends on the state of the “Global Default” setting for the “vpconf startup” profile.
  - If Global Default is enabled:
    - Since no vpconf profile is configured for new aircraft, the Global Default (“startup”) profile is pushed to the device.
      - Since the startup vpconf was pushed with spring gain = %75, we read and update our stored gain settings from the device
  - If Global Default is disabled
    - Since no vpconf profile is configured for the new aircraft and Global Default is disabled, the vpconf settings and stored gain values (spring = %80) from the previous aircraft will persist.

- TelemFFB pushes spring gain %75 if Global Default is enabled or %80 if Global Default is disabled. Both of these pushes are redundant since those gain values are already on the device, but this is the way the logic works to account for cases when there are gain overrides but no vpconf overrides.

- Exit TelemFFB
  - TelemFFB pushes original %50 that was read on startup as final measure to ensure same state as startup.

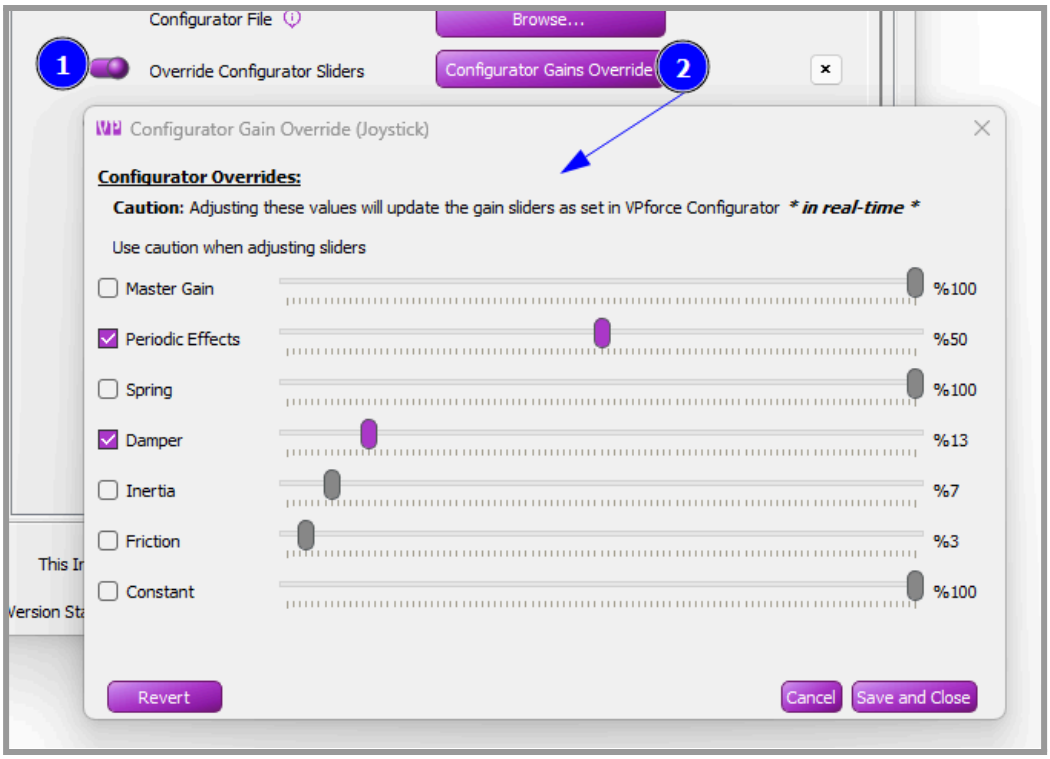
Configurator	Spring Gain		~~50%~~
TelemFFB	Startup vpconf spring gain		~~75%~~
TelemFFB	Configurator Spring Gain Override		80%
TelemFFB	Aircraft spring gain	x	40%
	Final Force		75%

4.12.2. Configuring The Gains

Configuring the Gain Overrides is very simple and similar to configuring the equivalent sliders in VPforce Configurator with one major difference. That is, adjusting the slider takes effect (very nearly) immediately. There is a small delay where the slider must be stationary in order for the command to be sent to avoid spamming the device with hundreds of commands.

To access the override dialog

1. Enable the “Override Configuator Sliders” toggle (1)
2. Press the “Configurator Gains Override” button (2)



To override a given effect gain slider, simply tick the checkbox and adjust the slider to your liking. You will feel the effects of the change immediately.

The button behavior is as follows

- **Revert Button**
  - The revert button will disable all of the override checkboxes and set the gains back to their stored baseline values. This will be either the gains read on startup or when the last “vpconf” profile was pushed
- **Cancel Button**
  - The cancel button will undo any changes that were made since the override window as opened. It will the close the window
- **Save and Close Button**
  - The save button will write the settings as they are currently configured to the user configuration file for the currently loaded aircraft.

4.13. Notable Sim Specific Info

This section expands upon some of the more complex or commonly adjusted settings in TelemFFB. A full list of all settings is available in the Effects Documentation Section

4.13.1. MSFS and X-Plane

4.13.1.1. Trim and Autopilot Following

TelemFFB supports trim and autopilot following in MSFS and X-planes, with special caveats for MSFS.

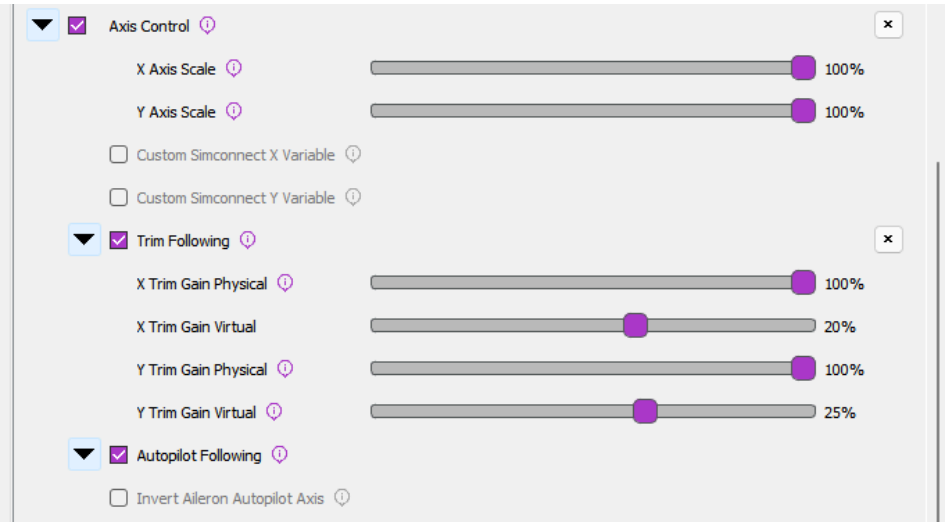
In order for TelemFFB to emulate movement of the joystick/pedals in response to trim or autopilot inputs, it needs to be able to control the axis position that MSFS is seeing from the joystick device. This is required since these simulators have no concept of FFB or axis offsets and will interpret any intentional deflection of an axis as ***deflection of the control surface*** and not just a response to the trim input. This is counteracted in software by limiting the amount of physical movement of the joystick that is actually communicated to MSFS.

For MSFS:

Since MSFS does not have a specific override toggle for external axis control, this means that in order to use this feature of TelemFFB, ***you must unbind your joystick or pedal axes inside of MSFS***. Otherwise, the internal joystick position will conflict with what is being sent by TelemFFB.

For X-Plane:

It is **not required to unbind your axes for X-Plane** since there are override toggles as part of the SDK. When the feature is enabled in TelemFFB, the axis is overridden.



To enable Trim and/or Autopilot following, simply enable the “Axis Control” feature and then in the sub-settings, enable Trim or Autopilot following accordingly.

- **Axis Scale:**
  - These sliders can be used to adjust the scale of the axis as sent to MSFS. A value of %50 will result in %50 control deflection in the sim with %100 physical deflection
- **Custom Axis Variables:**
  - Some aircraft do not use the standard simconnect events for their axes, or use custom LVAR variables. You can use these checkboxes to override the default variable that is sent or input a custom LVAR. Use "VARNAME" for simvars or "L:VARNAME" for LVARS
- **Trim Following Gains**:
  - X/Y Gain Physical
    - These gain settings affect how much the physical axis will move in response to the trim value. A gain of %100 will result in full travel of the physical axis with full travel of the trim
  - X/Y Gain Virtual

- These gain settings define how much of the physical movement is translated to the simulator over the simconnect session. A value of %50 means that only %50 of the physical movement of the axis will be sent to the simulator, resulting in the virtual axis moving %50 as much as the physical axis.

How it works at a high level:

- Trim Following
  - Trim position is read from the sim
  - Physical stick center point is calculated using the ‘physical’ position gain
  - Physical stick center is sent to the joystick/pedals
  - Virtual stick position is calculated using the ‘virtual’ position gain
  - Virtual stick position is sent to MSFS
- AP Following
  - Elevator AP following is reliant on the trim value, as APs use the elevator trim
  - Aileron/Rudder
    - Control surface deflection is read from the sim (as induced by AP control)
    - Control surface deflection is used to calculate physical axis position
    - Physical position is sent to joystick/rudder
  - The AP induced physical control inputs are dampened to prevent out of control oscillations in turbulence or in aircraft with extra sensitive controls.

Tips on configuring the trim settings

Physical & Virtual configuration should be done for each plane.

Suggested starting points:

```
X Gain Physical = 50%

X Gain Virtual = 20%
```

```
Y Gain Physical = 100%

Y Gain Virtual = 20%

Rudder Gain Physical = 50%

Rudder Gain Virtual = 20%
```

Joystick..X and Rudder..X can typically be left as default, since many planes do not even have in-cockpit trims on those axes, and if they do they are set and forget. The elevator trim however is interacted with a great deal and joystick..Y must be tuned per plane for realistic results.

Fly the plane, and trim for level flight at cruise speed.

In VPForce configurator, temporarily set spring to 0% and set friction to a value high enough that your stick stays in place when you let go of it. Apply (do not store) the setting.

Without moving the Rhino joystick, use your trim buttons/keys/axis to nose down the plane.

If the nose goes up, adjust **Y Gain Virtual** 10% higher.

If the nose goes down, adjust **Y Gain Virtual** 10% lower. It may be negative.

Adjust the trim and observe the reaction again. It will take a few iterations. The goal is to have the trim adjustment have no effect with the stick not moving. You can adjust by 5%, 1% when you are close.

Enjoy your new realistic trim!

4.13.1.2. Aileron/Elevator/Rudder Gain Settings

There are multiple ways the axis spring gains can be configured for aircraft in MSFS/X-Plane.

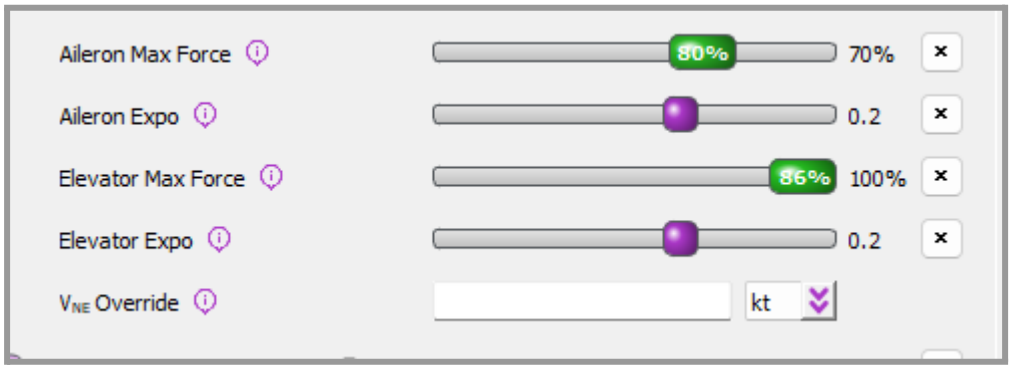
Dynamic - Spring gain changes based on increasing/decreasing dynamic pressure as airspeed changes, includes additional dynamic forces related to slip, AoA and g-forces.

Dynamic + Spring Centered - Adds a fixed gain centering force to the Dynamic spring effects. Where the standard Dynamic effect can reach 0 spring and 0 airspeed, the addition of the base centering force will set the lower boundary of the spring effect to the configured value

Fly By Wire (FBW) - Static spring force is configured per axis based on the settings.

4.13.1.2.1. Dynamic

There are settings which directly affect the max force per axis as well as an “exponent” setting which affects the curve at which the gain will be applied over the speed envelope of the aircraft.



Max Force Settings

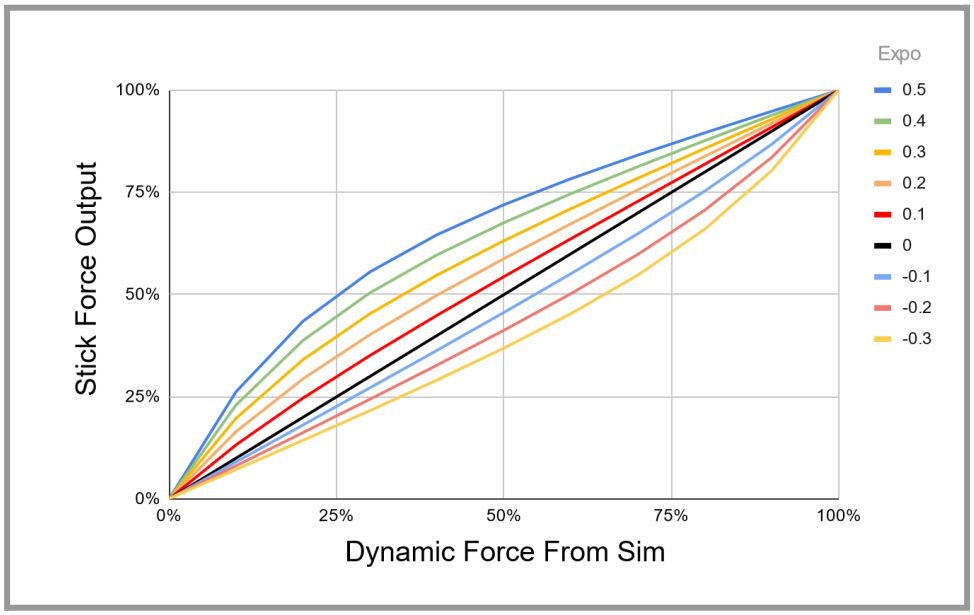
The “Max Force” settings will effectively set the spring gain that will be achieved at the V_{NE} (never exceed) speed of the aircraft, although the calculation is more sophisticated than a basic linear gain-to-speed mapping. It uses the known aircraft info to determine the dynamic pressure (Q) that should be achieved at V_{NE} for the aircraft and then feeds that information into the dynamic forces calculation to determine the final spring gain at any given point in time.

100% of the configured **Max Force** is achieved at the aircraft’s V_{NE} speed as read from telemetry. In the event that the V speeds defined in the aircrafts configuration files are incorrect, or if you want to override the value, it can be changed with the **V_{NE} Override** setting.

The Max Force adjustment slider handle will fade from gray to green as Max Force is reached, and the handle will show a percentage of dynamic force applied.

Expo Settings

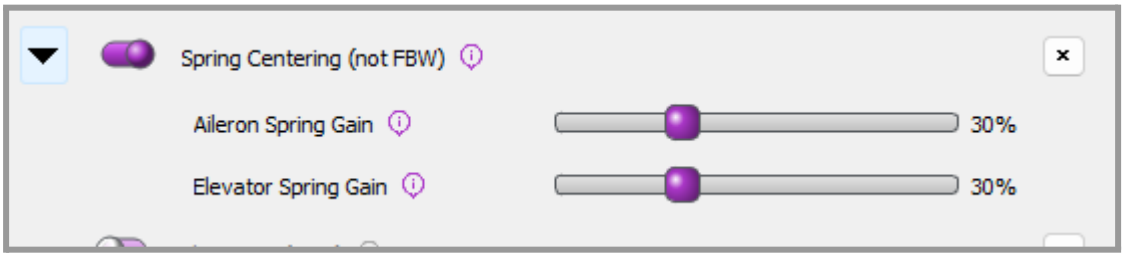
Since Rhino cannot produce the actual real-life forces that could be reached, Expo amplifies those forces at lower speeds, where the feeling of control authority is quickly lost at stall speeds for example. An Expo value of 0.5 doubles stick forces at 25% of V_{NE}. For some jets, you might want diminished forces until closer to V_{NE}, so you can set a negative Expo value.



4.13.1.2.2. Dynamic + Spring Centered

The “Spring Centered” option will still leverage the Dynamic adjustments mentioned above, however there will be a minimum spring gain set on a given axis based on the sliders.

With this configured, the dynamic spring gain will range from a low-point of the “Spring Centered” gain value to a high-point of the Max Force setting in the dynamic adjustment settings.



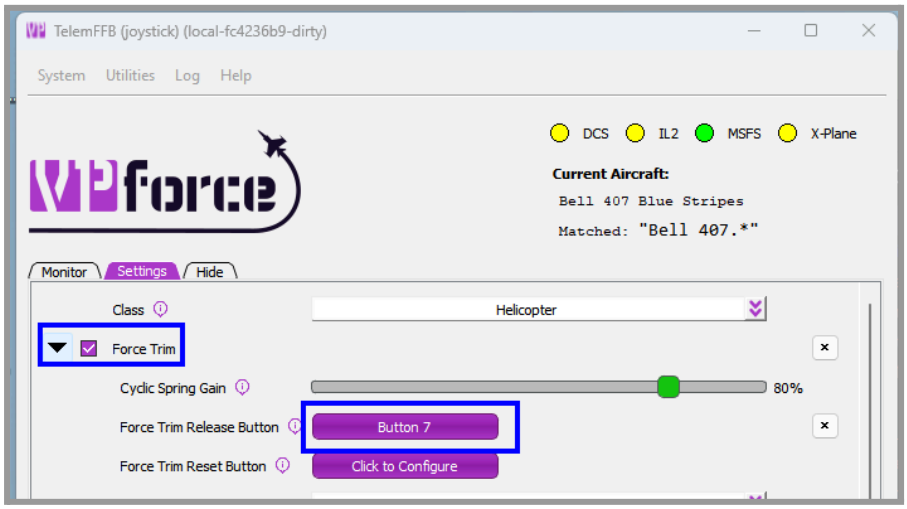
4.13.1.2.3. Fly By Wire (FBW)

Enabling the FBW option will override any configurations in the Dynamic and/or Spring Centered settings and apply a fixed gain value on a given axis. When this mode is active, the spring gain is static and will not vary based on airspeed or any other aerodynamic conditions.

4.13.1.3. Helicopter Force Trim

Helicopter force trim emulation is supported for both MSFS and X-Plane. To enable this feature of TelemFFB, enable the Force Trim checkbox and then in the sub-settings, configure a button on your joystick to serve as the trim release button.

Note: If you enable force trim, but do not set a button, you will see a  indiction for the simulator. The Trim Release button is mandatory, the Trim Reset button is optional.



4.13.1.4. (MSFS Only) - Special FFB Implementation for Hype Performance Group Airbus Helicopters

In collaboration with HPG, this implementation in TelemFFB was developed as a true-to-life representation of piloting the Airbus H145 and H160 aircraft.

The VPforce Rhino will work with the AFCS and act as the auto trim motor does, slowly moving the joystick as required to keep the SEMAs within their range of travel. The Rhino is also integrated with the force trim release system and the “hands on” spring override detection system. Force trim for hand-flying is also supported.

Both the Cyclic and Collective axes (if you have a VPforce powered collective) are integrated with the AFCS. The Tail Rotor axis is also supported.

Excerpt from the [HPG H145 user guide]:

AFCS (Autoflight System)

The H145 autopilot is a comprehensive autoflight system, capable of both basic stabilization and also fully-hands-off upper modes. The system combines redundant Stability Augmentation Systems with redundant aircraft management computers, which take data from aircraft sensors and send commands to the actuators. The systems are monitored by and interacted with through the MFDs, Autopilot control panel (APCP) and controls on the cyclic and collective.

Background

The H145 flight controls are augmented by parallel actuators, called SEMAs (smart electro-mechanical actuator). These parallel actuators are invisible to the pilot (not felt in the controls) and are controlled by aircraft computers directly. The SEMA are quick and powerful but limited in travel. The total SEMA travel will be only 10% of the pitch axis and 20% of the roll and yaw axis. For this reason, the AFCS also needs the ability to re-center the SEMA. The A.TRIM (auto trim) system is able to receive commands from the AFCS computers and then slowly drive the trim motor in the requested direction. As the trims move, the pilot sees and feels their cyclic moving. It is for this reason that the A.TRIM system must be engaged to use UPPER MODES, as otherwise the saturation of the SEMA could not be automatically resolved by the computers.

As part of this implementation, there are certain requirements and recommended settings in the MSFS control bindings, the HPG Helicopter settings (iPad) and in TelemFFB.

****NOTE** - Because of the unique aspects of this implementation, when either the H145 or H160 profiles are loaded, a series of aircraft specific L:Vars are subscribed to. These L:Vars are part of the default profiles for the H145 and H160 aircraft. As such, it is important that if you load a livery that does not match the default profiles, that you CLONE from the existing default profile. If you simply create a new entry of type "HPGHelicopter", it will not work properly.**

VPForce Configurator Settings:

- You must ensure that there is enough spring force enabled in the profile to properly center the joystick
- Joystick:
    - If the joystick sags away from center due to grip weight or low spring force:
        - use the ‘balance springs’ feature to counteract the grip weight
        - use the ‘adaptive centering’ feature to assist bringing the stick to center position when you are not holding it.
- Collective & Pedals:
    - In order to properly emulate AFCS control, spring force MUST be enabled on both the collective and the pedals

TelemFFB Settings:

- **Axis Control** must be enabled.
    - This is required for both the Cyclic axes and the Collective axis (if you are using a VPforce powered Collective)
    - You must UNBIND the axes in MSFS
- **Force Trim** must be enabled
    - you must also set your force trim binding in the force trim sub-configuration in TelemFFB
- **Cyclic**
    - **Hands-On Deadzone**
    - **Hands-Off Deadzone**
- **Collective**
    - **Collective AP Spring Gain**
    - **Collective Dampening Gain**

MSFS Settings:

- You must **UNBIND** your Cyclic axes in MSFS to prevent conflicts with TelemFFB sending the axis position

- If using a VPforce powered Collective,
    - You must **UNBIND** your Collective axis in MSFS to prevent conflicts with TelemFFB sending the axis position
    - You must **BIND** a button on your collective to act as collective trim release. Pressing the trim release is required to manipulate the real helicopters collective and that is modeled in TelemFFB. The binding in MSFS is `**AUTOTHROTTLE DISCONNECT**`

HPG H160/H145 Settings:

Depending on the version of the helicopter you have installed, the tablet options may differ. Use the tablet settings below depending on what your tablet options look like.

Older Versions:

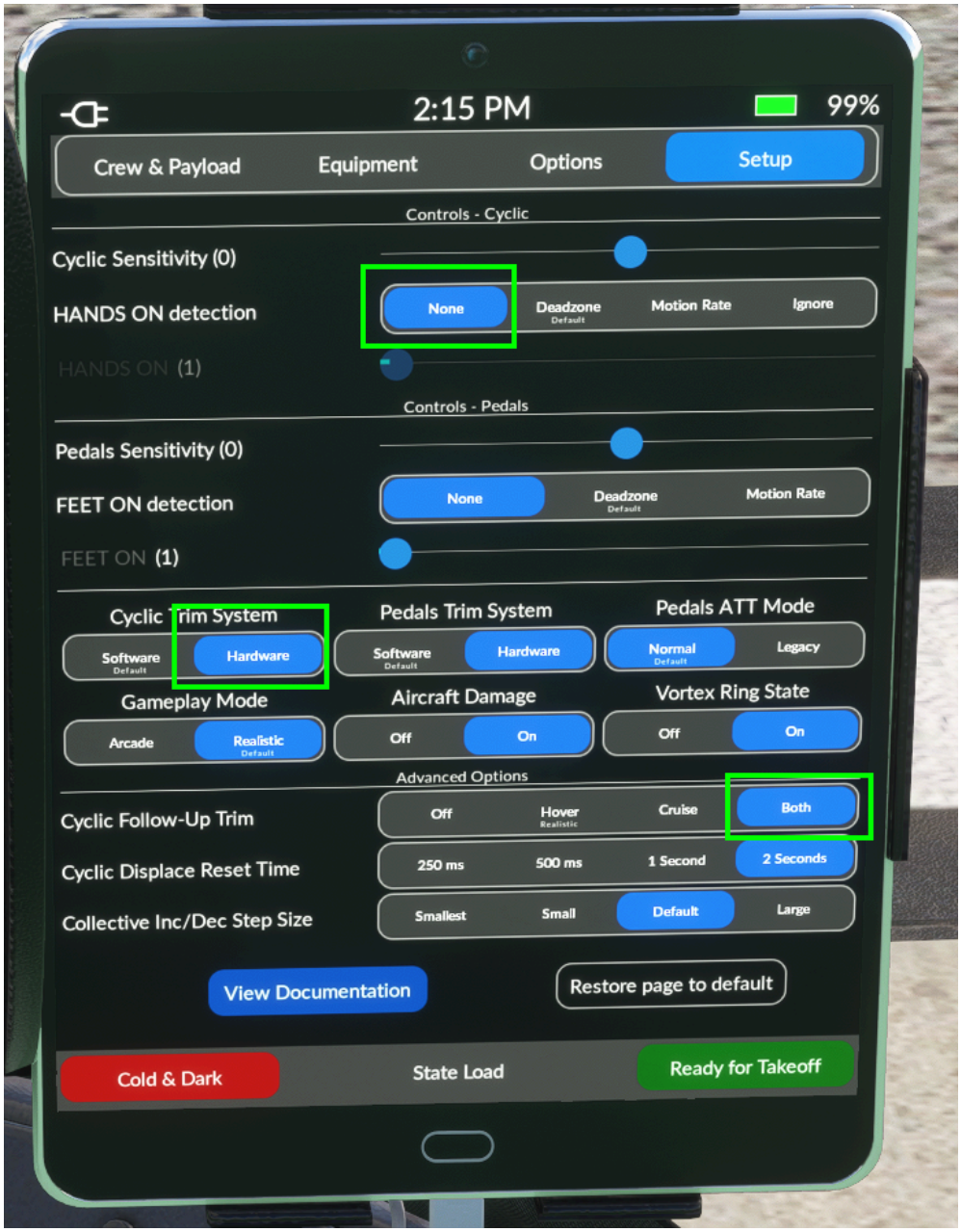
In the tablet settings inside the aircraft, the following must be configured for proper behavior:

- Cyclic:
    - Cyclic Control set to **'No Springs'**
    - Follow-Up trim set to **'OFF'** (you may need to temporarily enable Centering Springs to set this)
    - SAS Stability level
        - For the **H160**: between -80 and -60
        - For the **H145**: between -50 and -20
- Collective
    - SAS Stability level -100

Newer Versions:

Newer versions of the HPG helicopters have more options that assist with FFB implementations. You will want to set:

- Hands on Detection: **'None'**
- Cyclic Trim System: **'Hardware'**
- Cyclic Followup Trim: **'Both'**



4.13.1.5. Low Hydraulic Pressure Effect

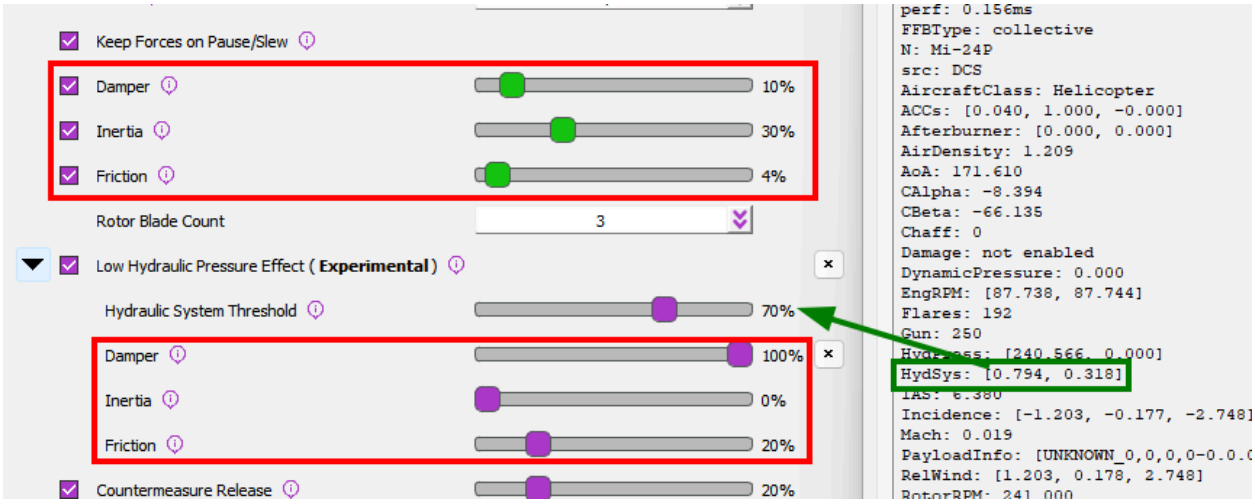
This effect allows the configuration of damper, inertia, and friction forces above and beyond those which are set by the base damper/inertia/friction settings in TelemFFB.

Note *In order for this effect to work, the Damper/Inertia/Friction effects must also be enabled.*

Note *Care must be taken when increasing these forces. Particularly with Inertia and Friction. Adding too much of these forces can quickly lead to motor instability issues, resulting in motor fault protection shutdown.*

*** Note*** *It is important to understand that all of these slider settings are limited by what is configured in the active VPForce Configurator profile. If your basic damper/inertia/friction forces are enabled at %100 in TelemFFB, there will be no room for the low pressure effect to increase them further.*

TelemFFB monitors the data in the "**HydSys**" telemetry and will linearly apply these effect values in place of the standard values between the threshold setting value and a 'HydSys" value of 0. If the HydSys value is a list, the effect uses the max value to determine whether or not the pressure is below the threshold.



When setting the Hydraulic System Threshold setting for a new aircraft, you must first determine what "normal" is, by inspecting the **HydSys** telemetry value under normal conditions. Then set the Hydraulic System Threshold slider to a value *less than* the normal operational

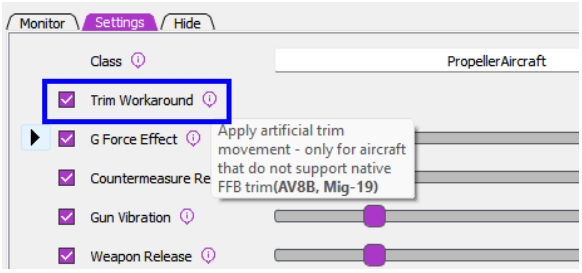
value. If the HydSys value drops below the threshold, the effects force settings will begin taking effect.

4.13.2. DCS

4.13.2.1. Joystick Trim Workaround

Some DCS modules do not properly implement joystick following for trim inputs. This feature mimics the trim movement by moving the physical joystick with the trim.

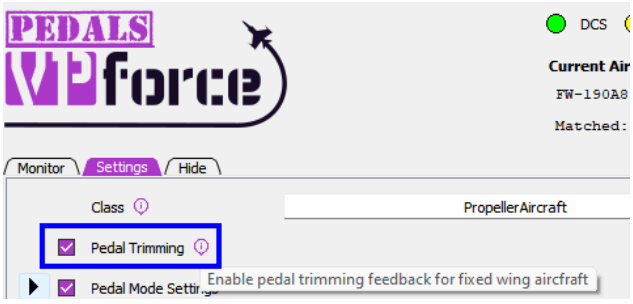
Note: When this setting is enabled, TelemFFB is overriding the axis position and sending it to DCS. As such, **the axis scaling, curve and inversion settings in DCS are overridden**



4.13.2.2. Pedal Trim Following

This setting will enable trim following for DIY Rudder pedals.

Note: When this setting is enabled, TelemFFB is overriding the axis position and sending it to DCS. As such, **the axis scaling, curve and inversion settings in DCS are overridden**

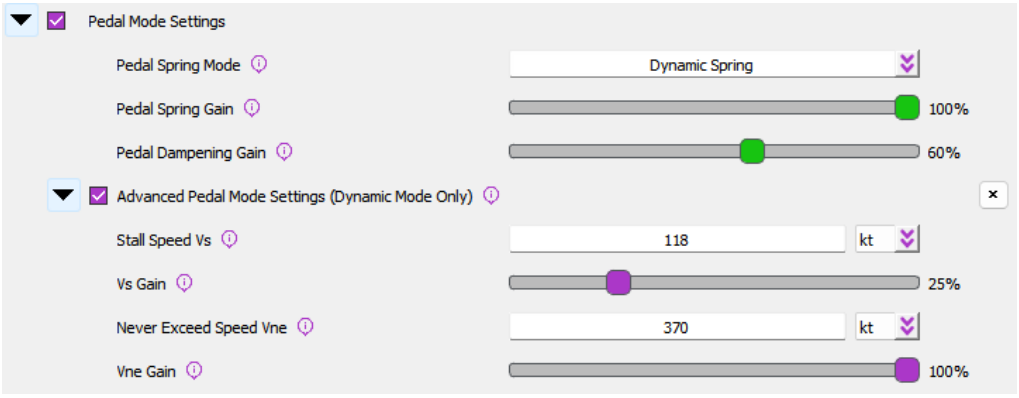


4.13.2.3. Pedal Mode

DCS does not natively support FFB pedals. TelemFFB has implemented basic FFB capabilities.

There are 3 pedal modes supported. Dynamic (default for warbirds), Static (default for jets) and No Spring (default for helicopters).

For Dynamic mode, there are advanced settings that are available for adjusting how the spring force is applied across the speed envelope of the aircraft.



4.13.2.4. Low Hydraulic Pressure Effect

See the documentation for this effect in the MSFS Low Hydraulic Pressure Effect Section above. The effect works largely the same way for DCS.

Support is currently limited to:

- UH-1, SA342, Mi-8, Mi-24, KA-50
- A-10C, AV-8B, F-14, F15ESE

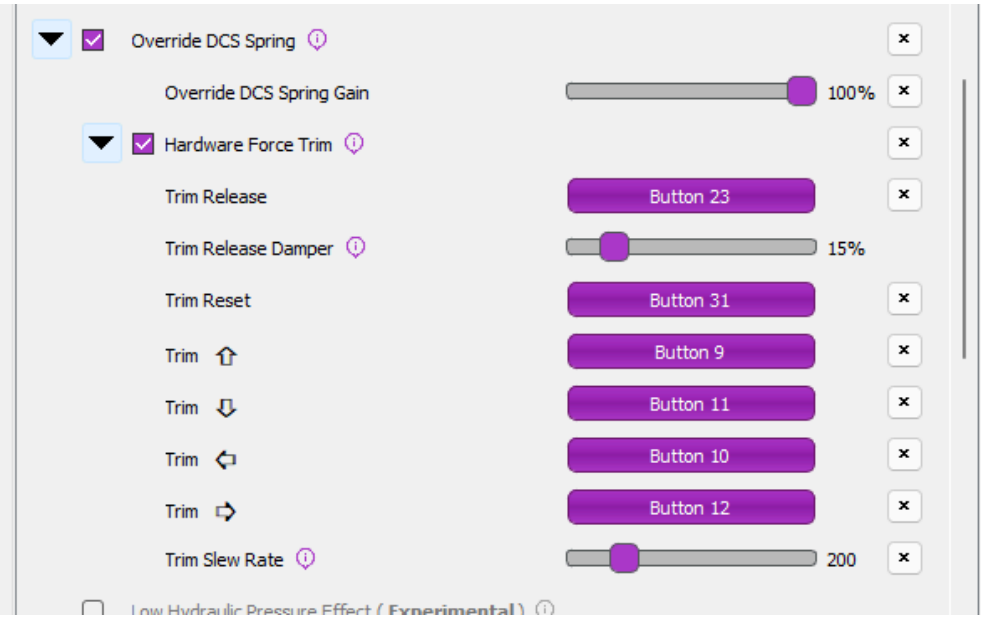
The primary difference is that for each DCS aircraft, the telemetry must be individually sourced in a unique way per aircraft. As such, the supported aircraft are limited at this time. See the TelemFFB release notes for the supported aircraft.

For DCS Aircraft, the Hydraulic System Threshold setting has already been coarsely configured for each of the supported aircraft, depending on how the data is being read and what the normal values are.

4.13.2.5. Override Spring options for DCS

In order to better support the popular community developed modules for DCS, that do not support FFB, functionality was added to TelemFFB which replicates the “sticky spring’ and hardware based trim functionality that can be accomplished via VPforce configurator.

Implementing this functionality in TelemFFB simplifies the enablement of spring force and trimming controls for users of these modules without having to resort to configuring and deploying dynamic configurator profiles.



When enabled, TelemFFB will start an “override spring” effect on the joystick that will supercede anything that is (or is not) set by DCS. The spring gain is configurable, but is still bound by the spring gain setting that is defined in the active profile on the device.

Hardware trim options are available for helicopter cyclic type force trim, a reset binding as well as 4-way hat trim with configurable slew rate.

The slew rate is in “steps per second”. 1 step is equal to 1/8192 of the total throw of the axis.

4.13.3. IL-2

4.13.3.1. Duplicate ‘Shake’ effects

IL2 implements FFB for dynamic stick forces and some very basic shake effects. TelemFFB implements duplicate (but far more configurable) effects which overlap with those that are implemented by IL2. To enable these specific settings, enable the “IL-2 Shake Master” setting in TelemFFB.

Note: It is recommended to set the “shake” intensity in the IL-2 FFB control settings to 0 if you enable these settings in TelemFFB.

4.14. TelemFFB and Antivirus Software

Why Antivirus Software May Flag This Application

This application is packaged using **PyInstaller**, a tool that bundles Python applications into standalone Windows executables. Occasionally, Windows Defender or other antivirus software may flag the generated `.exe` file as potentially malicious. This is a common issue across many open-source and independent software projects and **does not mean the application is unsafe**.

What Causes False Positives?

There are a few key reasons why antivirus software might misidentify the executable:

1. **Heuristic Scanning**
   Security suites often use heuristic analysis to flag behaviors typical of malware (e.g., dynamic imports, compressed binaries, network or file system access).
   PyInstaller-packaged apps often exhibit similar patterns due to how Python and its libraries are bundled.

2. **Bundled Dependencies**
   This app includes numerous open-source Python libraries, which are all extracted and compiled into a single executable. This results in a large and complex binary—sometimes resembling known malware in structure—especially when compression or UPX is used.

3. **Lack of Widespread Use or Code Signing**
   Applications that are new or not widely installed are more likely to be flagged.
   Additionally, because this application is not signed with a commercial code signing certificate, Windows may mark it as “unrecognized” or “unknown publisher,” increasing suspicion.

4. **Frequent Builds**

   Every build generates a slightly different binary (even without code changes), which antivirus vendors haven't yet seen. As a result, they may temporarily flag it until it's verified as safe by more users.

How We Ensure Safety

- All source code is openly available and auditable.
- Dependencies are widely used Python packages from the Python Package Index (PyPI).
- Builds are produced in a clean environment to prevent contamination.

What You Can Do

- **Allow the app manually** if it's flagged and you trust the source.
- **Submit the executable to Microsoft or your antivirus vendor** for review. This helps improve detection accuracy over time.
- **Check with VirusTotal** to independently verify whether the file is flagged across multiple engines.

4.15. Effects Documentation

This section attempts to document and explain each of the effects and their settings. It is a work in progress. A majority of the effects will apply to all simulators. Where applicable, each effect setting has a comment in-line with the default setting to indicate which simulator(s) it applies to.

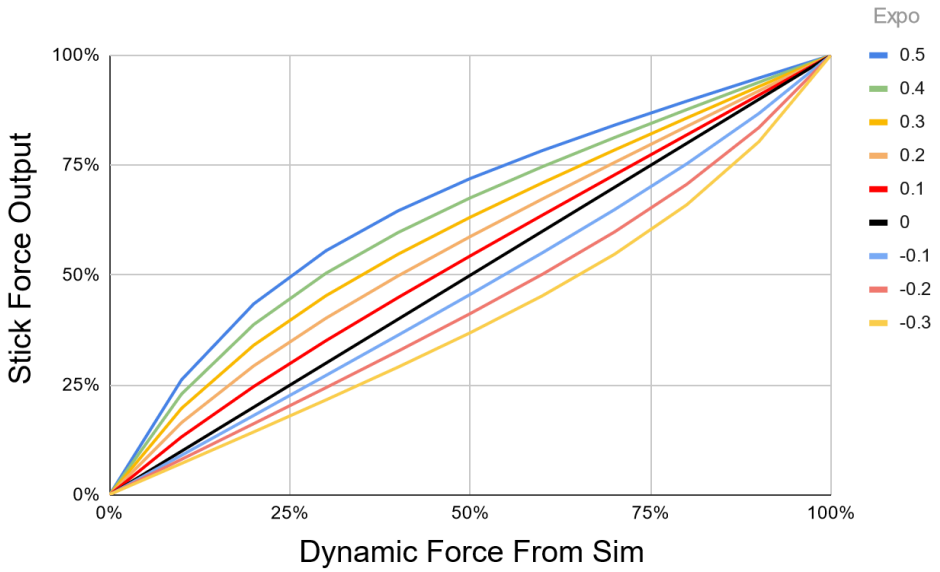
Dictionary of All Settings, in alphabetical order:

4.14.0.0.0.1. Afterburner Rumble

- DCS, MSFS, XPLANE Fixed Wing
- Afterburner rumble effect. Slider controls max intensity

4.14.0.0.0.2. Aileron Expo

- MSFS, XPLANE Fixed Wing
- Exponential value for use in dynamic airflow forces calculations. 100% of the set **Aileron Max Force** is achieved at the aircraft's V_{NE} speed read from telemetry, which may be changed with the **V_{NE} Override** setting. Since Rhino cannot produce the actual real-life forces that could be reached, Expo amplifies those forces at lower speeds, where the feeling of control authority is quickly lost at stall speeds for example. An Expo value of 0.5 doubles stick forces at 25% of V_{NE}. For some jets, you might want diminished forces until closer to V_{NE}, so you can set a negative Expo value.



4.14.0.0.0.3. Aileron Max Force

- MSFS, XPLANE Fixed Wing
- Aileron maximum scalar for use in dynamic airflow force calculations. The handle will fade from gray to green as Max force is reached, and the handle will show a percentage of dynamic force applied. Max Force is a percentage of the spring force value in VPforce Configurator, so your total stick force multiplier at any time is Configurator Spring % x Max Force (slider position) x Dynamic Force (% shown in handle)

4.14.0.0.0.4. AoA Effect

- DCS, MSFS, XPLANE Fixed Wing
- Enable or disable the dynamic Angle of Attack (AoA) based force effect
    - **AoA Effect Gain** Amount of calculated AOA effect to apply
    - **AoA Effect Max Force** Maximum constant force to apply for the AoA effect

4.14.0.0.0.5. AoA Reduction

- DCS, MSFS, XPLANE Fixed Wing
- Simulates the increased forward stick pressure that is applied on some fighter aircraft when a critical angle of attack is exceeded. The effect will monitor the AoA and apply a

linear force, up to the maximum defined value starting at the ‘start’ AoA and maxing out at the ‘max’ AoA. This is a percentage of the constant force value in VPforce Configurator

- **Critical AoA Max Speed** AoA at which applied force maxes out
- **Critical AoA Start Speed** AoA at which to begin applying force

4.14.0.0.0.6. AoA/Stall Buffeting

- DCS, MSFS, XPLANE Fixed Wing
- Peak AoA buffeting intensity

4.14.0.0.0.7. Autopilot Following

- MSFS, XPLANE Fixed Wing
- Requires **Axis Control** and **Trim Following** - Enable physical stick following of autopilot movements. ****Autopilot following is only available for fixed-wing aircraft and HPGHelicopters at this time****

How it works**:**

- Elevator AP following is reliant on the trim value, as APs use the elevator trim
- Aileron/Rudder
  - Control surface deflection is read from the sim (as induced by AP control)
  - Control surface deflection is used to calculate physical axis position
  - Physical position is sent to joystick/rudder
- The AP induced physical control inputs are dampened to prevent out of control oscillations in turbulence or in aircraft with extra sensitive controls.

Invert Aileron Autopilot Axis If aircraft becomes unstable with AP Following and wants to flip inverted, try this option

4.14.0.0.0.8. Axis Control

- MSFS, XPLANE
- The trim/autopilot following features requires that TelemFFB be the source of the axis position for MSFS. As such, the following settings will enable and configure the sending of the axis positions via simconnect.

****Note You must un-bind your axes in MSFS or SPAD.next for this feature to work**

The input range used by MSFS is -16383 to +16384. Axis curves are not supported in this implementation, however you can utilize the scaling settings to adjust the sensitivity of the physical axis. A (unreasonable) scale value of %50 would send a range of -/+8192 over the full range of the physical axis, resulting in less sensitive control inputs at the expense of range of movement.

X Axis Scale Scaling of axis position sent to game, 0-100%.

Y Axis Scale Scaling of axis position sent to game, 0-100%.

Rudder (X) Axis Scale Scaling of axis position sent to game, 0-100%.

The following simconnect events are used to send the axis position data:

Fixed Wing (or with **Use Legacy Bindings** in Helicopter Classes):

```
AXIS_AILERONS_SET

AXIS_ELEVATOR_SET

AXIS_RUDDER_SET
```

Helicopter:

```
AXIS_CYCLIC_LATERAL_SET

AXIS_CYCLIC_LONGITUDINAL_SET

ROTOR_AXIS_TAIL_ROTOR_SET
```

4.14.0.0.0.9. Buffet Onset Aoa

- DCS
- AoA when buffeting starts

4.14.0.0.0.10. Canopy Motion

- DCS
- Peak vibration intensity when canopy is moving

4.14.0.0.0.11. Center on Pause/Slew

- MSFS, XPLANE
- Force spring centering when in pause/slew mode When disabled, you will need to bring the axis close to center to re-establish axis control

4.14.0.0.0.12. Class

- Aircraft Type, can be one of PropellerAircraft, TurbojetAircraft, GliderAircraft, JetAircraft, Helicopter, or HPGHelicopter. Choices based on sim availability.

4.14.0.0.0.13. Collective AP Spring Gain

- MSFS, XPLANE Heli
- Defines the strength of the spring force to use when the collective trim release button is NOT pressed. See Special HPG Helicopter Implementation section

4.14.0.0.0.14. Collective Dampening Gain

- MSFS, XPLANE Heli - Defines the strength of the dampening effect to apply to the collective axis when the collective trim release is pressed See Special HPG Helicopter Implementation section

4.14.0.0.0.15. Command Runner

- Execute a shell command when the aircraft loads. Can be used to kick off another process, execute a batch script or any other action executable as a shell command.

4.14.0.0.0.16. Configurator File

- Load a specific .vpconf file (created with VPForce Configurator) When using this function, it will remain active until set again by another sim, class, or model setting.

4.14.0.0.0.17. Co-Pilot/RIO Spring Override

- DCS
- With this feature you can temporarily override the spring when moving away from the pilot seat. Optionally, you can confine the functionality to a button that must be held.

4.14.0.0.0.18. Countermeasure Release

- DCS
- Peak intensity for countermeasure release effect

4.14.0.0.0.19. Damage Effect

- DCS, IL2
- Plays a short random direction, random intensity bump each time damage is detected on the aircraft. **** Can potentially cause performance impact due to large number of calculations required in export script for some aircraft **** Written in a way that will only execute the code in the export script if the feature is enabled in TelemFFB. Note that with the randomized nature of the intensity, some hits will be lower and some higher than the defined value

4.14.0.0.0.20. Deceleration Effect

- DCS, IL2
- Monitors the deceleration G-forces on the aircraft and, if the aircraft is on the ground will apply a forward force (away from pilot) equal to the deceleration G-force up to, but not exceeding **Deceleration Max Force**. Deceleration effect pulls the stick forward when on the ground and decelerating

4.14.0.0.0.21. Elevator Droop Moment

- MSFS - Strength of elevator droop at rest (pushes stick forward)

4.14.0.0.0.22. Elevator Expo

- MSFS, XPLANE Fixed Wing
- Elevator expo value, See **Aileron Expo**

4.14.0.0.0.23. Elevator Max Force

- MSFS, XPLANE Fixed Wing
- Elevator maximum value, see **Aileron Max Force**

4.14.0.0.0.24. Elevator Prop Flow

- MSFS, XPLANE Fixed Wing
- Scaling of dynamic airflow effects on elevator

4.14.0.0.0.25. ETL Effect

- DCS, MSFS, XPLANE Heli
- Enable Effective Translational Lift Shimmy
    - **ETL Start Speed** speed at which the ETL effect will start - m/s
    - **ETL Stop Speed** speed at which the ETL effect will stop - m/s

4.14.0.0.0.26. Flaps Motion

- Peak vibration intensity when flaps are moving

4.14.0.0.0.27. FlyByWire (FBW)

- MSFS, XPLANE
- Identifies aircraft as Fly-By-Wire. No airflow forces will be felt. ****Do not use together with Spring Center (not FBW)**** Gains are a percentage of the spring force value in VPforce Configurator
    - **FBW Aileron Gain** Fixed spring gain for FBW aircraft.
    - **FBW Elevator Gain** Fixed spring gain for FBW aircraft.
    - **FBW Rudder Gain** Fixed spring gain for FBW aircraft.

4.14.0.0.0.28. Force Trim

- MSFS, XPLANE Gliders & Helis

- Many gliders have a lever actuated trim positioning system that recenters the elevator trim to hold the control stick where the lever is released. Some helicopters have a similar function. Configure the buttons to use for Trim Release and optionally Reset. The implementation is identical to how hardware trim works when configured inside VPforce Configurator, however the benefit of doing it inside TelemFFB is that it is dynamically enabled when loading a glider or helicopter.
    - **Cyclic Spring Gain** - (Helicopters) - Percent of VPforce Configurator spring value (0-100%)
    - **Aileron Force Trim** - (Gliders) - Enable force trim on aileron axis
    - **Elevator Force Trim** - (Gliders) - Enable force trim on elevator axis
    - **Force Trim Release Button** - Button # to hold to release spring while moving axis. For HPG Helicopters, see section Special HPG Helicopter Implementation
    - **Force Trim Reset Button** Button # to recenter spring trims (optional)

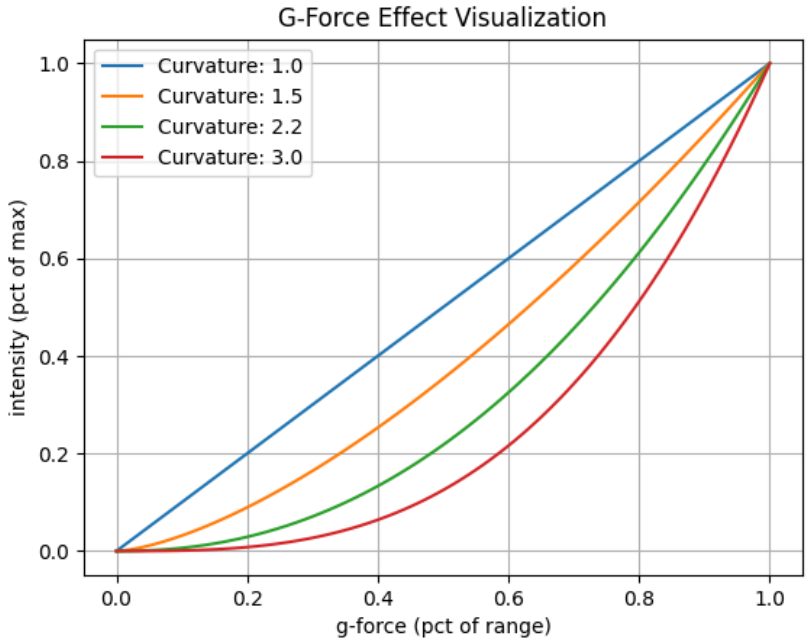
*4.14.0.0.0.29. G Force Effect *New**

- DCS/IL2/XPLANE/MSFS Fixed Wing
- Unlike the "legacy" g-force effect, this new version of the effect does not use an expo curve to calculate the output force. Rather, a force is calculated based on the current g loading as it exists between the min and max G settings. The physical stick deflection is then used to determine how much of this calculated force to apply at any given point in time.
- For example, if the current G loading is half way between min and max, that would result in %50 calculated force. If the stick is pulled back %50, this is factored with the original force value to determine the final output force (0.5 * 0.5 = %25). This happens in real time on every simulation frame. As you pull "harder", the g loading will increase, but the amount of the G loading which gets applied to the effect will also increase as the stick is pulled farther aft.
    - Maximum Intensity - This is the maximum force (as a percent of the configurator CONSTANT force slider) that will be applied
    - Start Gs - The G loading where the effect will start playing
    - Maximum Gs - The G loading where the strength will reach maximum value
    - Y Axis Max Point - Percentage of stick deflection that will result in %100 of the calculated force to be applied to the effect.
    - Enable Negative Gs - Enable the effect for negative G (<1.0)
        - Sub settings for this option are identical except the values will be negative

4.14.0.0.0.30. G Force Effect ("Legacy")

- The G-Force loading effect simulates the increasing force that is required to pull back on the stick as the G forces increase during a dive pull-out or hard turn. Slider value is a percentage of the constant force value in VPforce Configurator.
    - **Minimum Gs -** The G loading where the effect will start playing
    - **Maximum Gs -** G loading where the strength will reach maximum value
    - **G Force Curvature** - affects the onset characteristics of the force effect. A value of 1.0 is a linear increase in force across the defined g range. Increasing the curvature value will result in a flatter increase at the beginning of the range followed by an ever increasing force as the effect approaches the top of the range.

Example values (default is 2.2):



4.14.0.0.0.31. Gear Buffet

- DCS, MSFS, XPLANE
- Peak intensity for gear drag buffeting effect

4.14.0.0.0.32. Gear Motion

- Peak vibration intensity when gear is moving, and clunks at end of travel.

4.14.0.0.0.33. Gun Vibration

- DCS
- Peak intensity for gunfire effect

4.14.0.0.0.34. Hands Off Deadzone

- MSFS HPG Helis
- Distance at which hands-off resumes (MUST be lower than hands-on) See Special HPG Helicopter Implementation section

4.14.0.0.0.35. Hands On Deadzone

- MSFS HPG Helis
- Distance required to trigger a hands-on condition See Special HPG Helicopter Implementation section

4.14.0.0.0.36. Heli Engine/Rotor Rumble

- DCS, MSFS
- Rumble intensity for helicopter engine/rotor effect

4.14.0.0.0.37. IL2 Shake Master

- IL2
- While the majority of the settings for use in IL-2 are similar or identical to those that are used in DCS and MSFS, there are several that differ.
- IL-2 has several native FFB effects that can overlap with what TelemFFB is capable of generating. Specifically weapons release, runway rumble and buffeting. The benefit of implementing these in TelemFFB is that each effect is individually configurable both from an enable/disable perspective as well as the intensity.

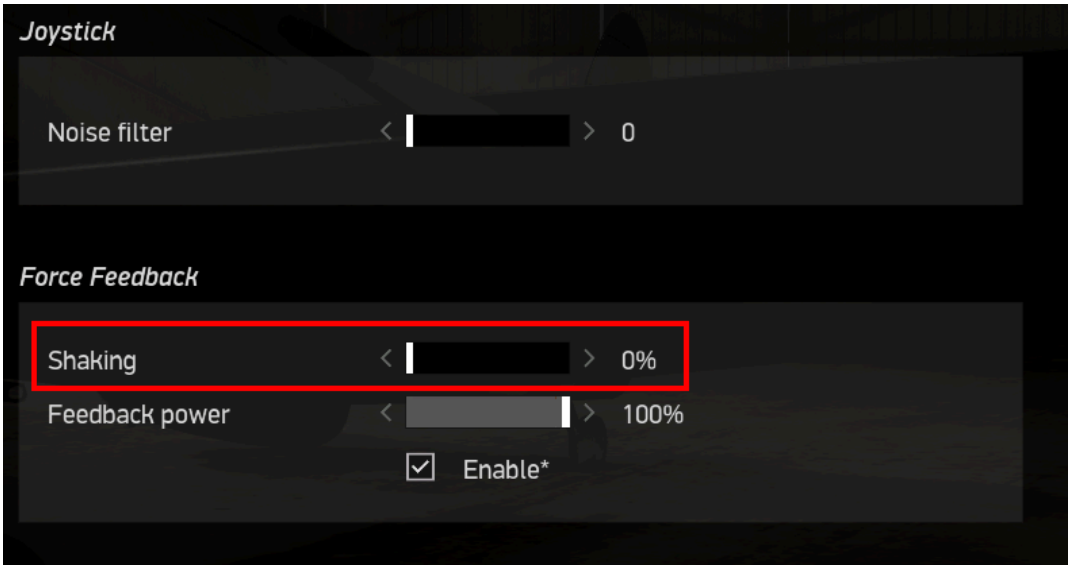
Buffeting - Common setting for all buffeting (stall, gear, etc) - (IL2 limitation)

Runway Rumble

Weapon Effects

These three effects are generated by IL-2 by default. If you wish to use the effects generated by TelemFFB in lieu of those generated by IL-2, you can enable the master setting in the TelemFFB config and then disable the ‘shaking’ effects in the IL-2 FFB

settings as follows. Navigate to Settings->Input Devices and move the 'Shaking' slider to 0.



4.14.0.0.0.38. Jet Engine Rumble

- Set intensity of jet engine rumble effect.
    - **Jet Engine Rumble Freq** Vibration Frequency

4.14.0.0.0.39. Nosewheel Shimmy

- Sets intensity of the nosewheel shimmy effect
    - **Nosewheel Shimmy min brakes**
    - **Nosewheel Shimmy min speed**

4.14.0.0.0.40. Overpeed Shake

- Sets the shake intensity when overspeed occurs
    - **Overpeed Shake Start Speed**

4.14.0.0.0.41. Override Configurator Sliders

- This feature allows you to override the gain sliders that are currently set via VPforce Configurator when loading into an aircraft. See the dedicated section Dynamic Configurator Gains for details.

4.14.0.0.0.42. Override DCS Spring

- **DCS**

- Allows to override the spring forces set by DCS and apply a static spring force. Useful for 3rd party mods that do not support FFB.

4.14.0.0.0.43. Pedal Spring Mode

- **DCS, IL2**
- In addition to pedal trimming, TelemFFB now implements dynamic switching between 3 different modes for Helicopters (No Spring), Jets (Static Spring) and Prop aircraft (Dynamic Spring). The modes may be overridden on a per aircraft basis by adding the applicable mode setting to that aircraft section in the configuration. In Dynamic Spring mode, force is based on Pedal Spring Gain, between 0 and Vs speed (%25 of force) and Vs and Vne speeds (remaining %75)

  All of the DCS warbirds have default values built into the application for the V speeds. It is possible to override the default internal V_S and V_{NE} speeds as well as the spring gains.

  Pedal Spring Gain Percent of spring setting in VPForce Configurator

  Pedal Dampening Gain Percent of damper setting in VPForce Configurator

  To change the V speeds or add V speeds to a non-warbird type aircraft, or adjust the gain values, you can edit the following settings in **Advanced Pedal Mode Settings**:

  - Stall Speed Vs
  - Vs Gain
  - Never Exceed Speed Vne
  - Vne Gain

4.14.0.0.0.44. Pedal Trimming

- **DCS**
- DCS does not properly support FFB pedals. As such, the following implementation has been added to TelemFFB to enable both correctly behaving spring forces as well as trimming for fixed wing aircraft that have rudder trimmers. Helicopter trimming is not currently supported as there are currently no viable methods to deal with the “double input” effect that is generated by the “instant trim” option for those helicopters which support pedal trimming. Additionally, helicopters like the Mi-24 implement an

approximation of the real helicopter's “foot microswitch” logic which detects when the pilot's feet are on the pedals. None of the modes for this simulation of that switch logic are conducive to integrating with FFB trim following.

The shining light is that with the auto-switching to springless mode for helicopters, pedal trimming is not really necessary. Default is ON for Propeller and Jet aircraft.

4.14.0.0.0.45. Prop Diameter

- MSFS, XPLANE
- Aircraft Prop Diameter, used in dynamic airflow calculations

4.14.0.0.0.46. Propeller Rumble

- Enable Propeller Rumble
- The two RPM and intensity settings work together to define how the effect behaves. At the **Low RPM** value, the rumble effect will be played at **Low Intensity**. As the RPM increases, the intensity will **decrease** proportionally all the way up to the **High RPM** value, where the intensity will reach **High Intensity**. Note that these are not floor values. If the RPM drops below **Low RPM**, the intensity will increase above **Low Intensity**.
- Generally speaking, high frequency vibrations will feel stronger at equal intensities. The “High RPM” intensity should be **lower** than the “Low RPM” intensity
    - **Engine Rumble High RPM** - high RPM threshold
    - **Engine Rumble High Intensity** peak intensity of engine rumble at high RPM
    - **Engine Rumble Low RPM** low RPM threshold
    - **Engine Rumble Low Intensity** peak intensity of engine rumble at low RPM

4.14.0.0.0.47. Rotor Blade Count

- Count of helicopter rotor blades, used in ETL and Heli Rumble.

4.14.0.0.0.48. Rudder Expo

- MSFS, XPLANE Fixed Wing
- Rudder expo value. See **Aileron Expo**

4.14.0.0.0.49. Rudder Max Force

- MSFS, XPLANE Fixed Wing
- Rudder maximum value. See **Aileron Max Force**

4.14.0.0.0.50. Rudder Prop Flow

- MSFS, XPLANE Fixed Wing
- Scaling of dynamic effects on rudder

4.14.0.0.0.51. Runway Rumble

- Peak runway rumble intensity

4.14.0.0.0.52. Speedbrake Buffet

- Peak intensity for speed brake buffeting effect

4.14.0.0.0.53. Speedbrake Motion

- Peak intensity for speed brake motion effect

4.14.0.0.0.54. Spoiler Buffet

- Peak buffeting intensity when spoilers deployed

4.14.0.0.0.55. Spoiler Motion

- DCS F14 Only
- Peak vibration intensity when spoilers are moving

4.14.0.0.0.56. Spring Centering (not FBW)

- MSFS, XPLANE
- Enable spring centering for aircraft while maintaining dynamic forces. ****Do not use together with FBW****. Gains are a percentage of the spring force value in VPforce Configurator

  Aileron Spring Gain - Aileron spring gain

  Elevator Spring Gain Elevator spring gain

  Rudder Spring Gain - Rudder spring gain

4.14.0.0.0.57. Stall AoA

- **DCS, IL2**
- Stall Angle of Attack

4.14.0.0.0.58. Trim Following

- MSFS, XPLANE

- Requires **Axis Control** - Enable physical stick movement with in-game trims.
- How it works:
    - Trim position is read from the sim
    - Physical stick center point is calculated using the ‘physical’ position gain
    - Physical stick center is sent to the joystick/pedals
    - Virtual stick position is calculated using the ‘virtual’ position gain
    - Virtual stick position is sent to MSFS

X Trim Gain Physical - Aileron/Cyclic Lateral physical movement scalar (100%)

X Trim Gain Virtual - Aileron trim movement of virtual (in-game) controls

Y Trim Gain Physical - Elevator/Cyclic Lon. physical movement scalar (100%)

Y Trim Gain Virtual - Elevator trim movement of virtual in-game controls. ****Out of these six settings, this is perhaps the most important one, since elevator trim is so commonly used**** Adjust Y Trim Gain Virtual so that moving the elevator trim in-game does not pitch aircraft when the stick is not physically moved. It helps to set spring force to 0 in Configurator (just Apply, do not Store) to find the best value for this parameter, because you don’t want any motion of the stick while trimming. To configure, fly aircraft straight and level at cruise speed. Keep the stick in one position and slowly apply nose-down trim. if the nose goes up, raise the virtual % - if the nose goes down, lower the virtual % - the value might be negative. You are aiming for no nose-up or nose-down with trim input. When you have found the value, return your spring force in Configurator to your previous value. You will find when flying the plane, the trim now simply “relieves the pressure” as it would in a real plane.

Rudder Trim Gain (physical) - Rudder physical trim movement scalar (100%)

Rudder Trim Gain (virtual) - Rudder trim movement of virtual (in-game) controls

4.14.0.0.0.59. Trim Workaround

- DCS
- Some DCS modules do not properly implement joystick following for trim inputs. This feature mimics the trim movement by moving the physical joystick with the trim

4.14.0.0.0.60. Uncoord Turn Effect

- MSFS
- Simulate body acceleration effect on stick in uncoordinated turns

4.14.0.0.0.61. Use Legacy Bindings

- MSFS Only
- For helicopters that still use aileron/elevator or rudder bindings
- Requires **Axis Control**

4.14.0.0.0.62. VNE Override

- MSFS, XPLANE Fixed Wing
- Overrides V_{NE} read from telemetry for dynamic force calculations

4.14.0.0.0.63. VRS Effect

- MSFS, XPLANE Helis
- Vortex Ring State effect

4.14.0.0.0.64. Weapon Effect Direction

- DCS
- Direction of applied force for weapons effects | range 0-359 or set to -1 for random

4.14.0.0.0.65. Weapon Release

- DCS
- Peak intensity for weapons release effect, 0 to disable

4.14.0.0.0.66. Wind Effect

- DCS
- Adjust the maximum intensity the wind effect can attain

5. Game Specific FFB Settings, Tips and Tricks

This section consists basically of a massive amount of collective knowledge in the community, from years of getting or trying to get force feedback to work in a variety of simulators. A lot of it concerns generic force feedback devices, but Rhino specific instructions are included where applicable.

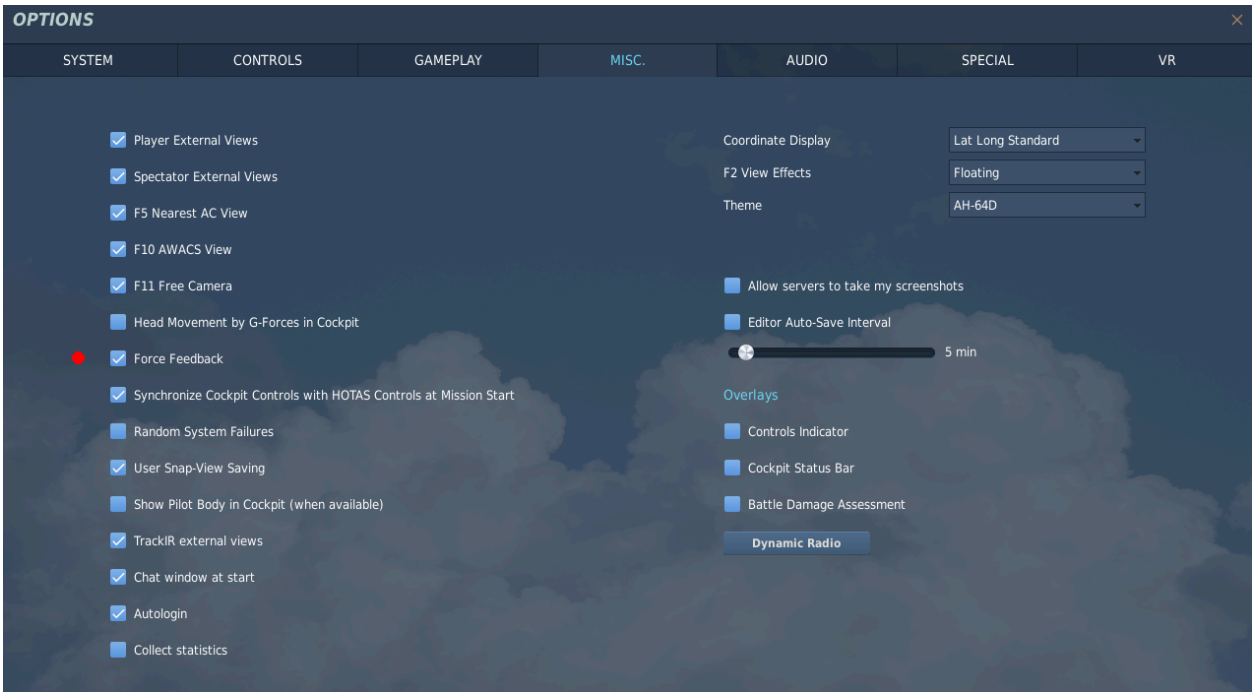
This section is expected to be updated somewhat regularly when it inevitably becomes necessary.

5.1. DCS World

DCS World is easy to get started with, because it offers native force feedback support through DirectInput. It's impossible to define a general level of force feedback support, though, because it varies so much based on the module. Generally the helicopters have force trim, warbirds model control stiffness from airflow (try getting one of your elevators shot off and enjoy the pleasantly light control feel), high AoA shake and weapons effects and the modern stuff varies from basically nothing at all to pretty good effects.

DCS offers fairly good data export, enabling projects like the TelemFFB to tap into the simulator and enhance the FFB experience - see the relevant section in this manual for more information.

The basic setup for Force Feedback is simple. Go to Options - Misc and enable Force Feedback as in the image below:



And that's pretty much it. The real work is in the modules, though, so it's not quite that simple.

If you have axis curves or saturation settings in the DCS axis tune configs, you will need to disable those as both curves and saturation are incompatible with FFB.

DCS FFB Support varies based on aircraft types:

- **Warbirds**
    - Flight surface pressure depending on speed (stick is limp on ground, pressure builds up during takeoff)
    - Stall effect (buffeting)
- **Cold War Era**
    - Stick center offset (F-5E, A10C, F-14 ...)
    - Stick offset through trimming
    - Autopilot features take full control over the stick (Mig21 ...)
- **Helicopters**
    - Full force trim (stick offsets and hold in position)
    - Trim button released means forces are applied
    - Trim button depressed means forces are off (for maneuvering)
- **Modern Jets:**
    - standard stick functionality (spring effect)

5.1.1. General note on DCS Helicopter Force Trim settings

All of the DCS helicopter modules support force trim in one fashion or another. For *all* helicopters “Special” settings menu, you will want to choose the “**Instant**” or “**Default**” trimming option for the cyclic.

The settings names are somewhat confusing however and prior experience with the “instant” trimmer and non-ffb joysticks often leads new FFB users down the wrong path.

Without FFB enabled in DCS and with no FFB joystick, the “instant” trim mode results in an immediate doubling of the input when the trim button is released. This behavior changes when FFB is enabled.

On many helicopters there is also a setting that is usually called “**Joystick with no springs and FFB**”. The actual intent of this setting seems to vary from helicopter to helicopter. For most helicopters it does not result in the desired behavior and the developers seem to have interpreted the meaning as “with no springs and with no ffb”. In only one case (the KA-50) does it seem to behave the same as the “instant” mode.

However, in all cases, the “instant” or “default” trimmer modes will work with FFB.

Note that DCS does not support FFB pedals, even if many of the helicopter modules have pedal trimmer settings in the special menu. None of the pedal modes for any of the helicopters has been found to be of any use for FFB.

5.1.2. Setup tips for DIY FFB Pedals and/or Collective devices

DCS does not natively support any FFB devices other than joysticks. While you can connect and use other FFB devices (like pedals), DCS does not understand what these devices are and does not differentiate them from the X/Y axes that are part of the joystick. DCS will

broadcast all “FFB events” to all connected FFB devices with the assumption that they all behave like joysticks with x/y axes tied to the control surfaces.

What this means is that if your pedals (or collective) are connected to DCS and you adjust the aileron trim in an aircraft, the spring offset that is generated for your joystick will *also* be sent to your pedals X axis. Similarly for a collective type device, the Y axis conflicts with the Y (pitch) axis when it comes to receiving FFB events from DCS.

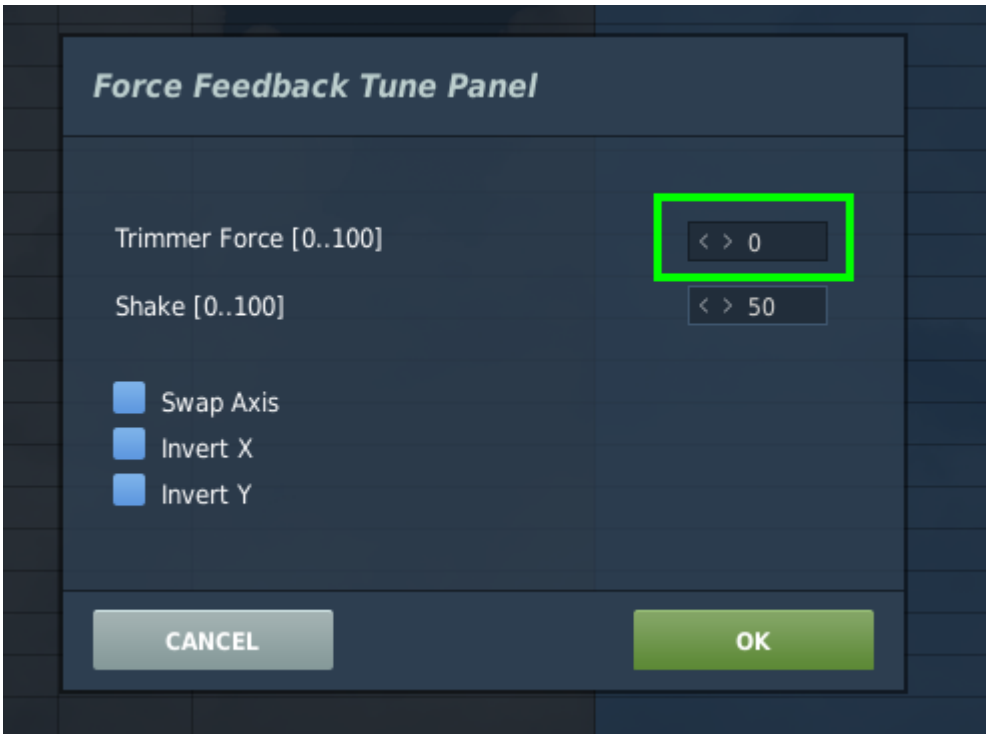
The VPforce TelemFFB application adds some basic pedal support for DCS (see section on TelemFFB).

The solution to this is to disable FFB trim forces for those devices in the DCS **“FF Tune”** menu. Many people don’t know these settings exist in DCS.

You can disable the FFB spring force per device, per aircraft in DCS. To do so:

- Enter the controls configuration for the aircraft and ensure that “**Foldable view**” is **unselected**.
- Select the column header for your pedals/collective
- Then select the ‘**FF Tune**’ button.
- In the resulting window, change the “Trimmer Force” value to 0 and apply
- Repeat for additional FFB devices and aircraft configurations.

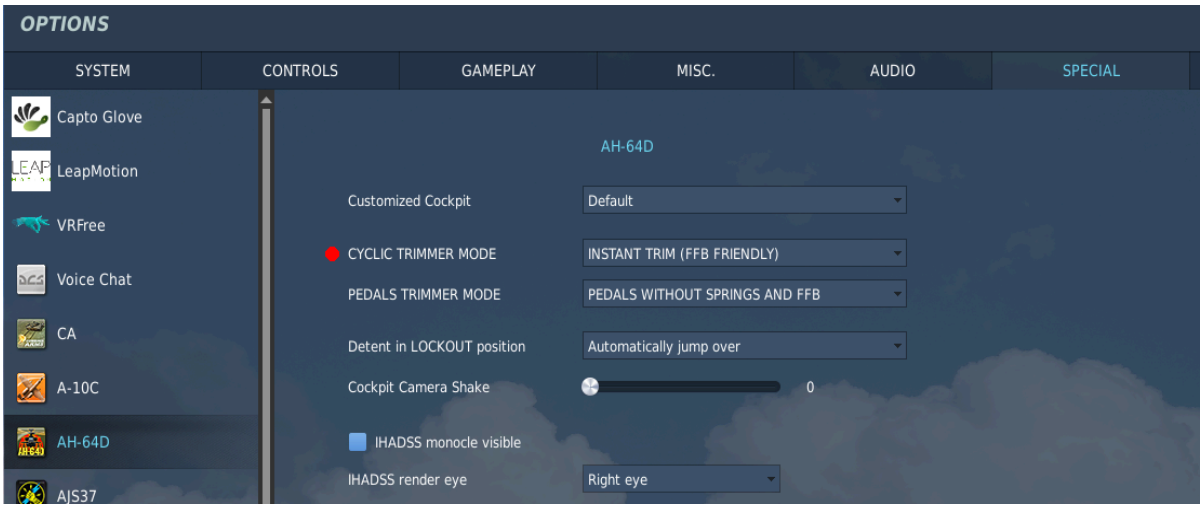

OPTIONS
SYSTEM
CONTROLS
GAMEPLAY
MISC.
AUDIO
SPECIAL
VR
F/A-18C
All But Axis Commands
Foldable view
Reset category to default
Clear category
Clear all
Load profile
Save profile as
Action
VPC Panel #2 (96E73C50-...
VP...
VPforce Rhino FFB Joystick...
VPforce Rhino FFB Pedals (433E17B0-0EE...
vJoy Device (2D2EA160-D...
LEFT VPC Throttle MT-50C...
Button Box Interface (6EE
A/A refueling - "Ready for precontact" radio call
AAU-52 Altimeter Pressure Setting Knob - CCW/Decrease
AAU-52 Altimeter Pressure Setting Knob - CW/Increase
Aircraft Bank Left
Aircraft Bank Right
Aircraft carrier catapult camera
Aircraft carrier catapult crew camera
Aircraft Labels
Aircraft Pitch Down
Aircraft Pitch Up
Aircraft Rudder Left
Aircraft Rudder Right
ALQ-165 ECM Mode Switch - BIT
ALQ-165 ECM Mode Switch - CCW
ALQ-165 ECM Mode Switch - CW
ALQ-165 ECM Mode Switch - OFF
ALQ-165 ECM Mode Switch - REC
ALQ-165 ECM Mode Switch - STBY
ALQ-165 ECM Mode Switch - XMIT
ALR-67 BIT Pushbutton
ALR-67 DIS TYPE Switch - A
ALR-67 DIS TYPE Switch - CCW
ALR-67 DIS TYPE Switch - CW
ALR-67 DIS TYPE Switch - F
ALR-67 DIS TYPE Switch - I
ALR-67 DIS TYPE Switch - N
ALR-67 DIS TYPE Switch - U
ALR-67 DISPLAY Pushbutton
ALR-67 DMR Control Knob - CCW/Decrease
ALR-67 DMR Control Knob - CW/Increase
ALR-67 OFFSET Pushbutton
ALR-67 POWER Pushbutton - OFF
ALR-67 POWER Pushbutton - ON
ALR-67 POWER Pushbutton - ON/OFF
ALR-67 POWER Pushbutton: ON<>OFF
ALR-67 SPECIAL Pushbutton
All Labels
All missiles padlock
Altitude 2-pos Switch: BARO<>RDR
Altitude Switch - BARO
Altitude Switch - BARO/RDR
Altitude Switch - RDR
AMPCD Contrast Control Switch - DOWN
AMPCD Contrast Control Switch - UP
AMPCD Gain Control Switch - DOWN
JOY_BTN12
JOY_BTN11
JOY_BTN23
JOY_BTN25
JOY_BTN37
JOY_BTN69
Modifiers
Add
Clear
Default
Axis Assign
Axis Tune
FF Tune
Make HTML
Disable hot plug
Rescan devices
CANCEL
OK

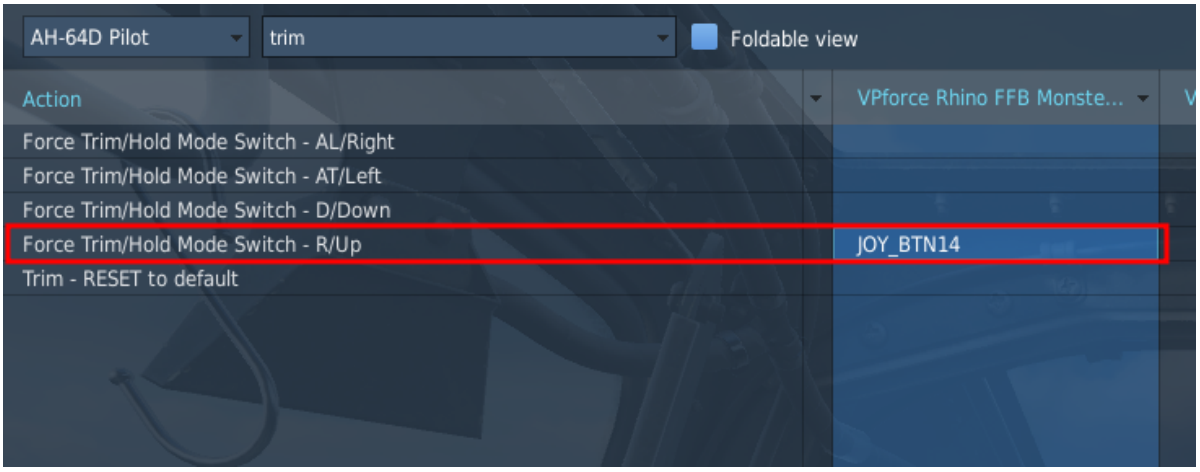

Force Feedback Tune Panel
Trimmer Force [0..100]
0
Shake [0..100]
50
Swap Axis
Invert X
Invert Y
CANCEL
OK

5.1.3. AH-64D Apache

The AH-64D Apache by Eagle Dynamics basically only offers support for force trim - but it is a very nice feature indeed and one of the best uses for force feedback in simulators so it's well worth having and the implementation in the AH-64D is good.

To enable FFB trim for the RHINO, go to Options - Special - AH-64 and set CYCLIC TRIMMER MODE to INSTANT TRIM (FFB FRIENDLY). For once, a feature is basically what it claims to be.





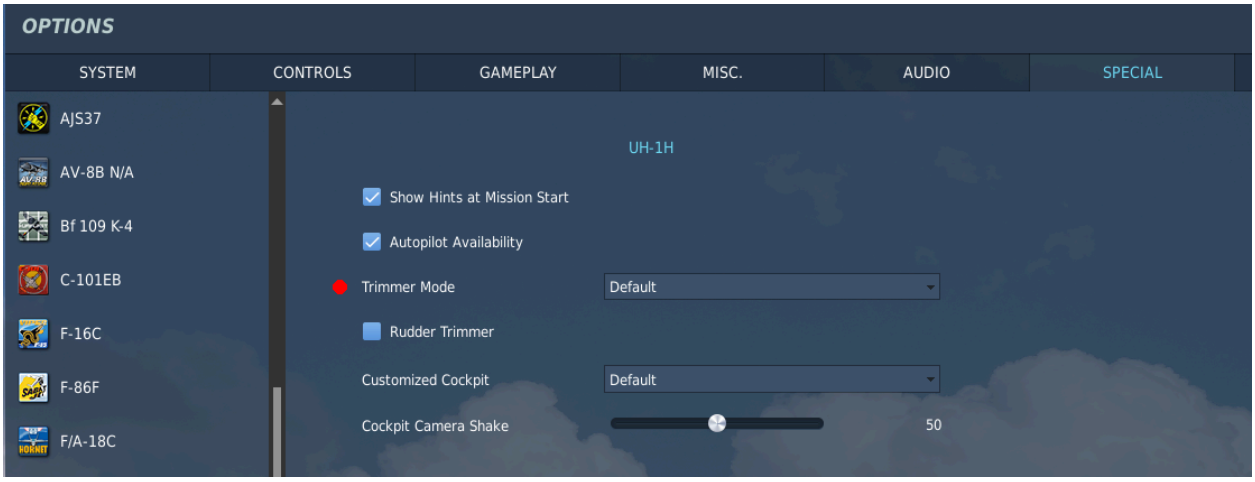
The PEDALS TRIMMER MODE option should be whatever best suits your hardware, it will have no effect on the RHINO.

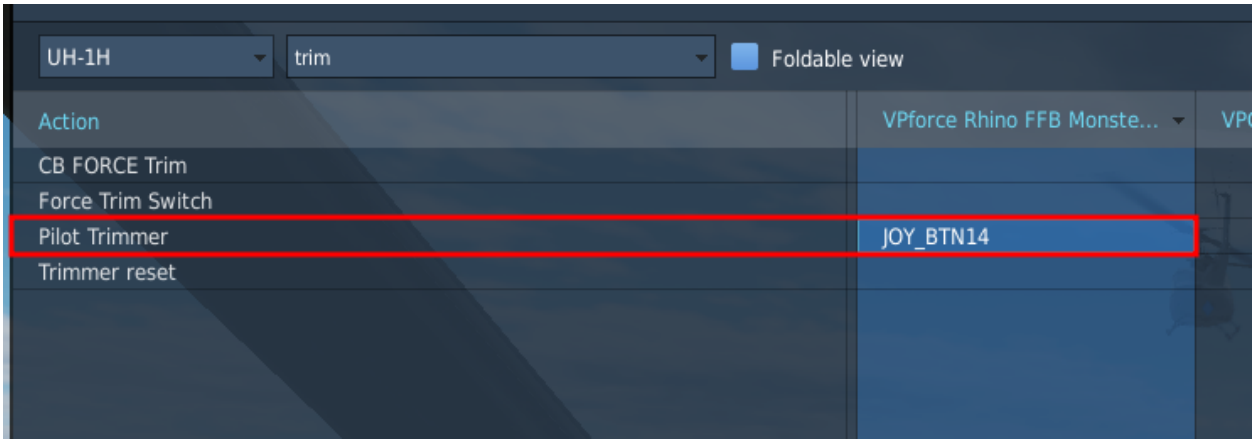
Note that as in many other helicopters in DCS World, anything that affects the in-game cyclic will affect the RHINO. It is highly recommended to do the initial testing and any troubleshooting in a scenario where the aircraft is guaranteed to be intact, properly configured and ready to go.

5.1.4. UH-1H Huey

The UH-1H Huey offers the always useful force trim feature, like most other DCS helicopters. The implementation is good and well worth the effort.

To enable FFB trim for the RHINO, go to Options - Special - UH-1H and set Trimmer Mode to Default as in the picture below. The other modes are designed for non-FFB controllers.



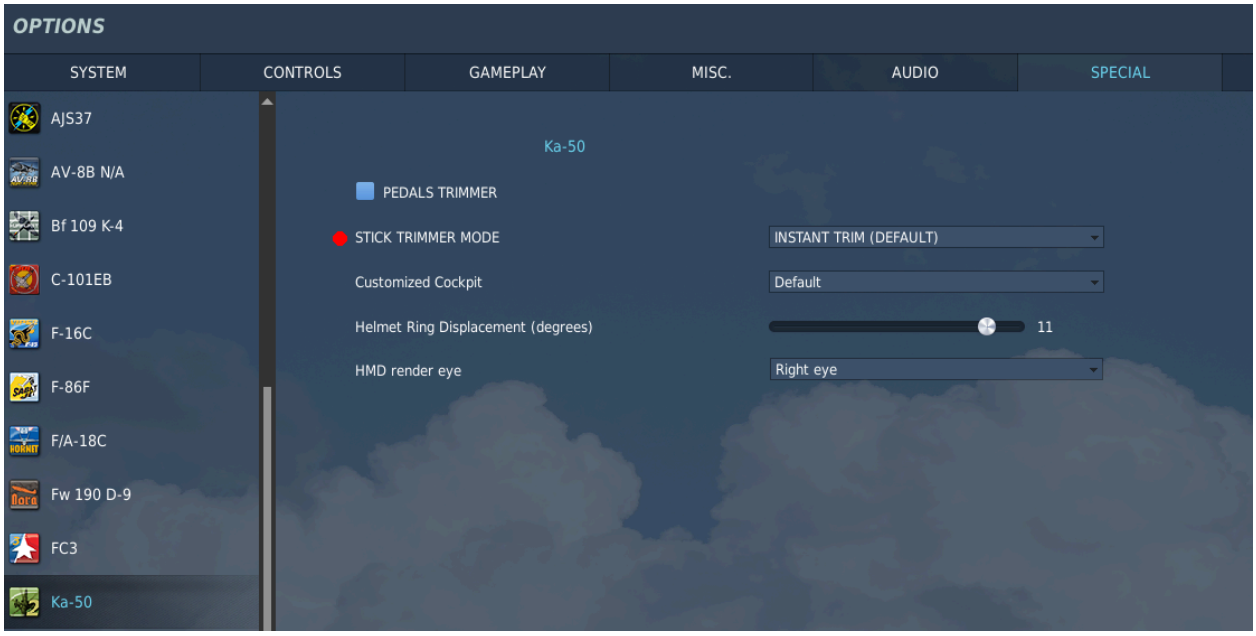


As with many other DCS helicopters, RHINO functionality is tied to the in-game cyclic functionality and if the aircraft is without power, configured incorrectly or broken, force trim won't work. Don't forget to turn on the force trim switch shown below:

It is highly recommended to do the initial setup or any troubleshooting in a scenario where the aircraft is guaranteed to be intact, correctly configured and ready to go.

5.1.5. Ka-50 Black Shark

As is typical to DCS helicopters, The Ka-50 offers force trim functionality, but no other effects. To enable force feedback for the Ka-50, go to Options - Special - Ka-50 and choose INSTANT TRIMM (DEFAULT) in the STICK TRIMMER MODE:

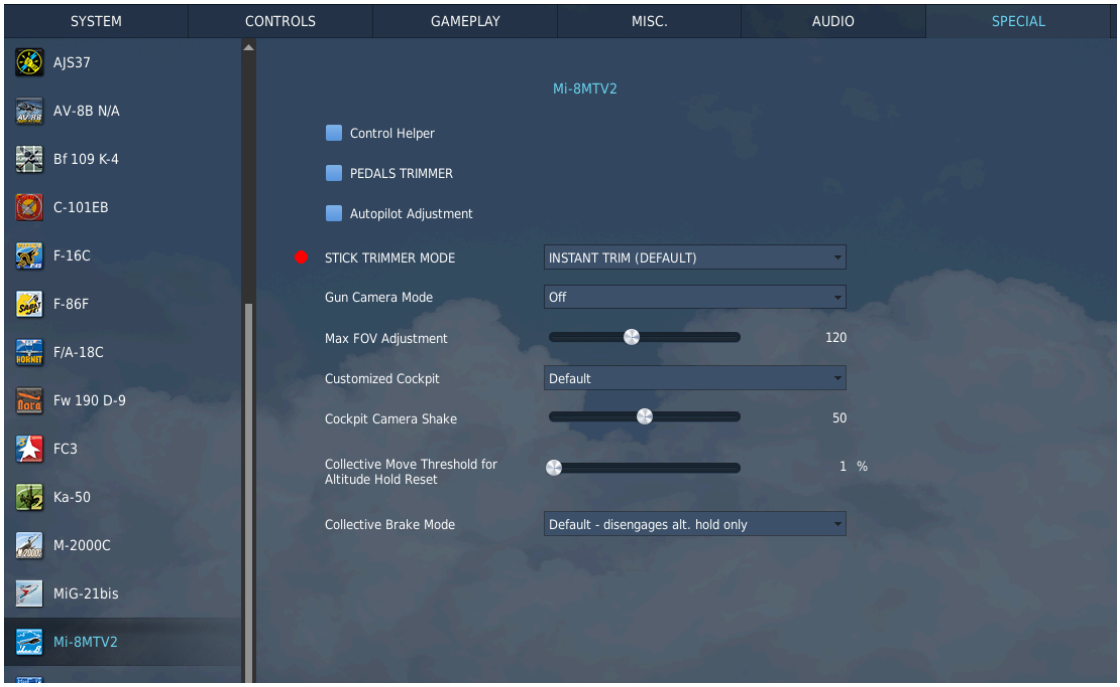


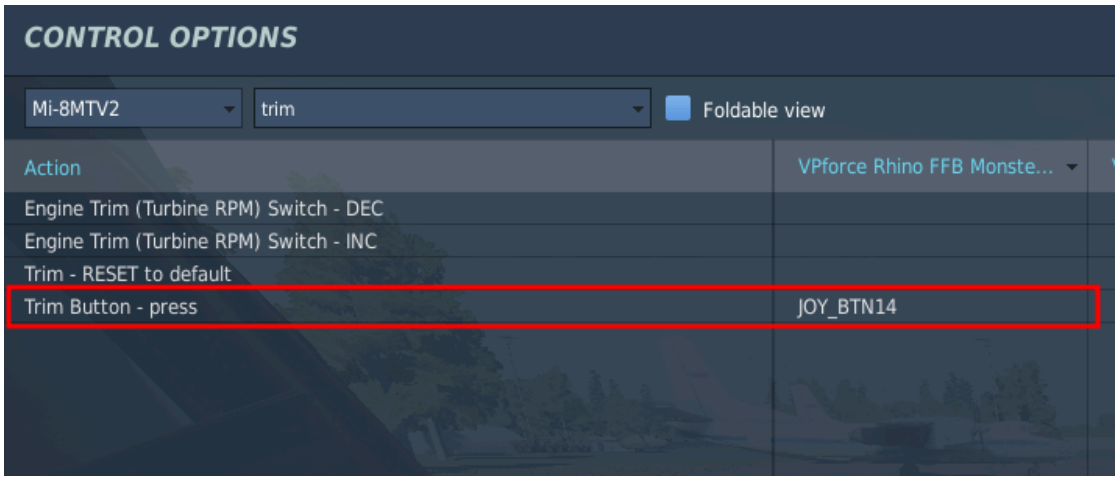
PEDALS TRIMMER has no effect on the Rhino. It seems that generally DCS helicopter trims default to correct force trim functionality and the extra options are different workarounds for non-FFB controllers.

Also typically, the Ka-50 force trim on the Rhino is dependent on what happens in the virtual cockpit, so make sure the aircraft is intact, powered up and ready to go before initial setup and testing.

5.1.6. Mi-8MTV2

Just like the other helicopters so far, The Mi-8MTV2 offers a well working force trim option, but no other effects. To enable force trim, choose INSTANT TRIM (DEFAULT) as the STICK TRIMMER MODE as in the image below:



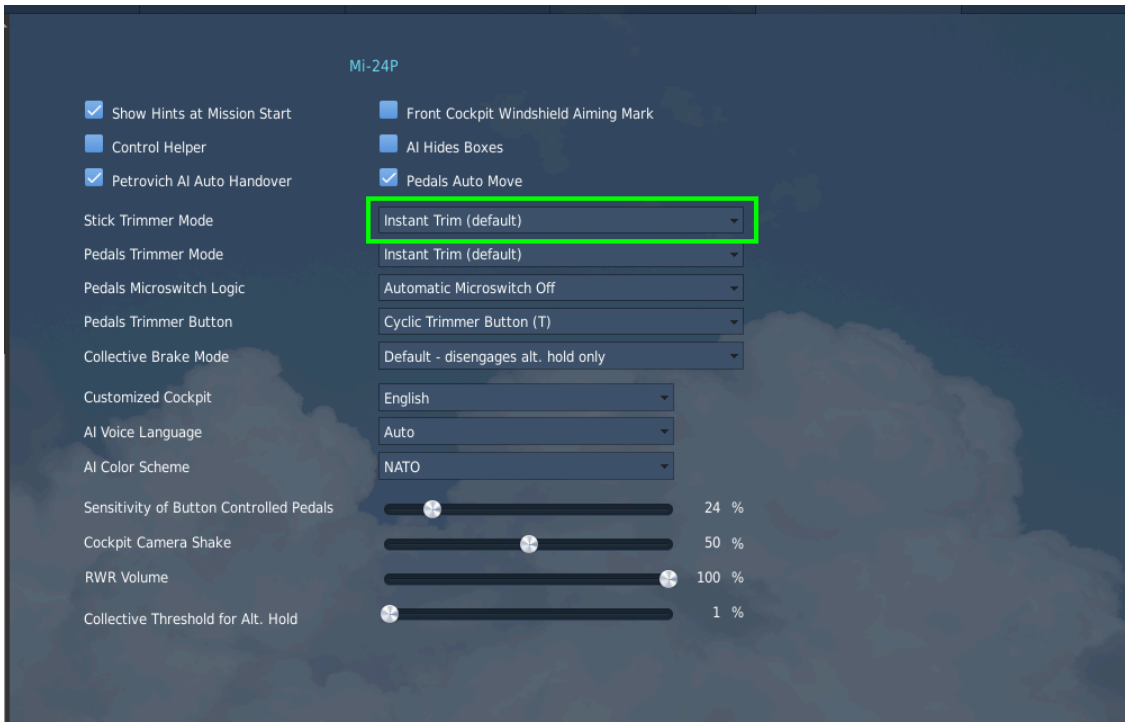


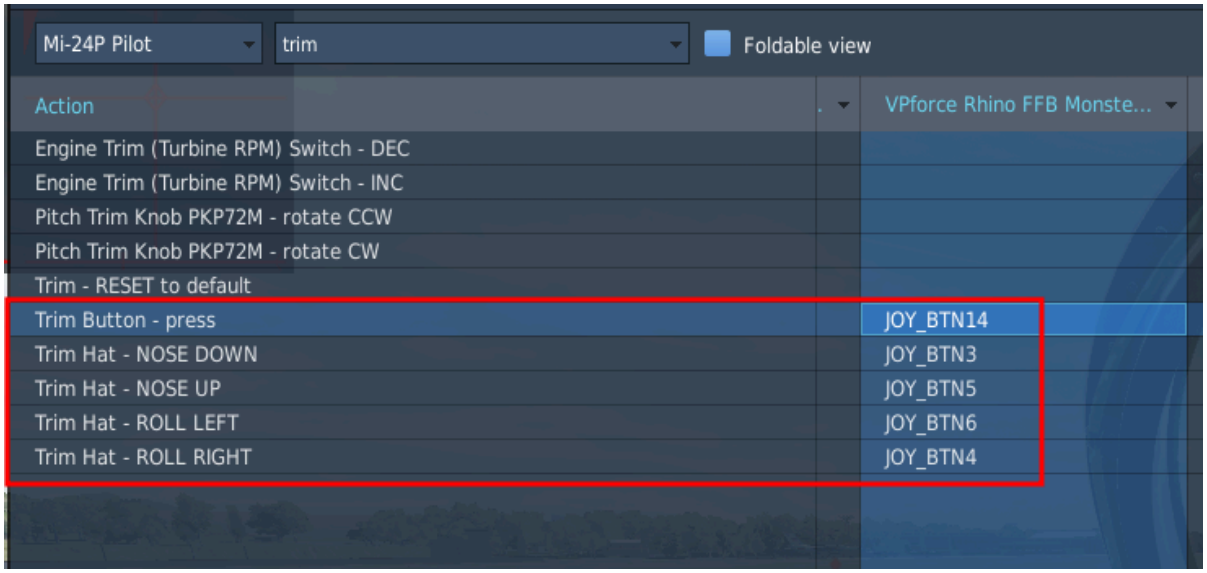
As usual, during the initial setup and for troubleshooting, make sure the aircraft is intact, powered up and ready to go. The Rhino force trim will mirror the in-cockpit one and won't work if the aircraft is not set up correctly

5.1.7. Mi-24P Hind

The Hind follows the same configuration logic as most of the other helicopters. In the special settings menu, be sure that the Stick Trimmer Mode is set to “Instant Trimmer (default)”

The Mi-24 supports “beep trim” trim following. Depressing the trim hat will cause the stick to move in the appropriate direction





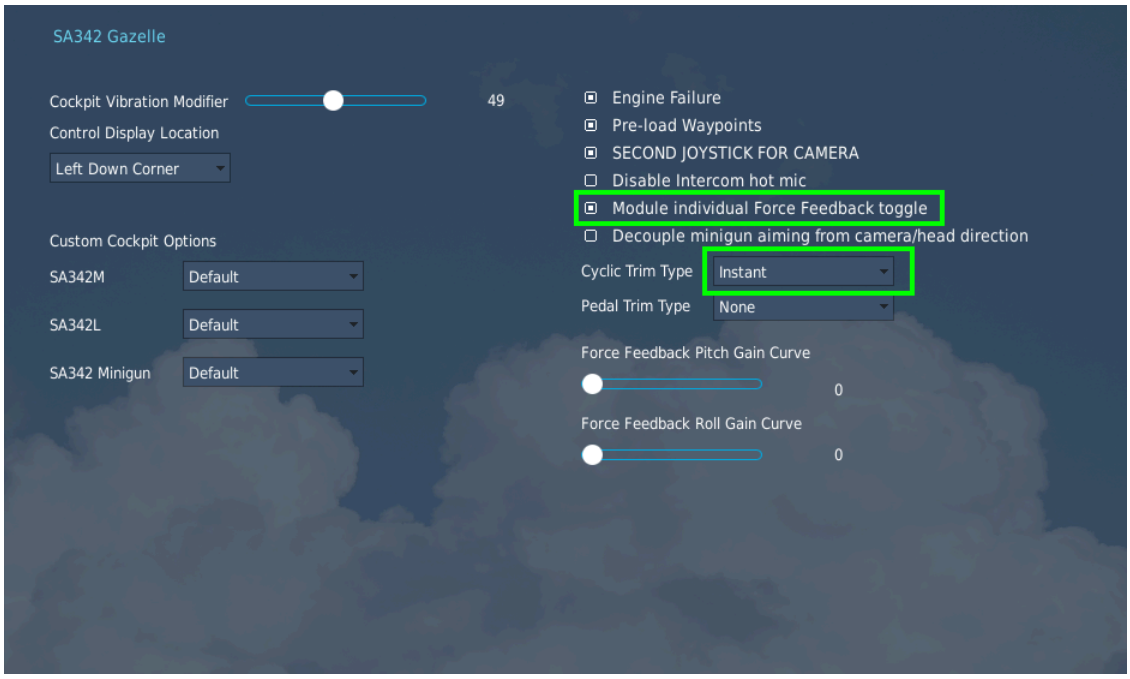
5.1.8. SA342 Gazelle

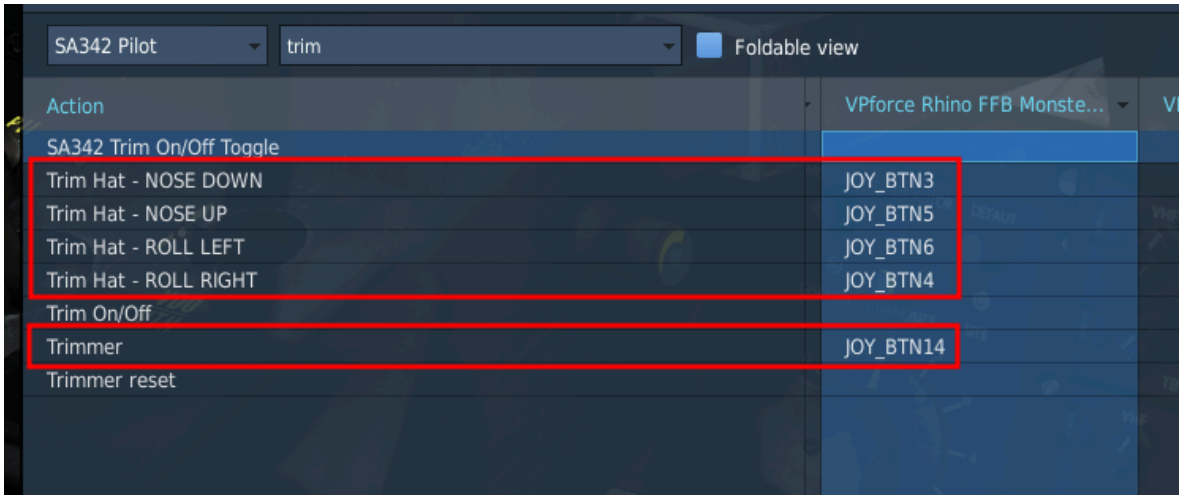
The Gazelle is the only helicopter which has its own FFB toggle in the special settings menu.

Enable the “Module individual Force Feedback toggle” setting and ensure the Cyclic Trim Type is set to “Instant”

The Gazelle supports “beep trim” trim following. Depressing the trim hat will cause the stick to move accordingly.

The trim on/off bindings control the trim actuator switch that will enable/disable beep trim.

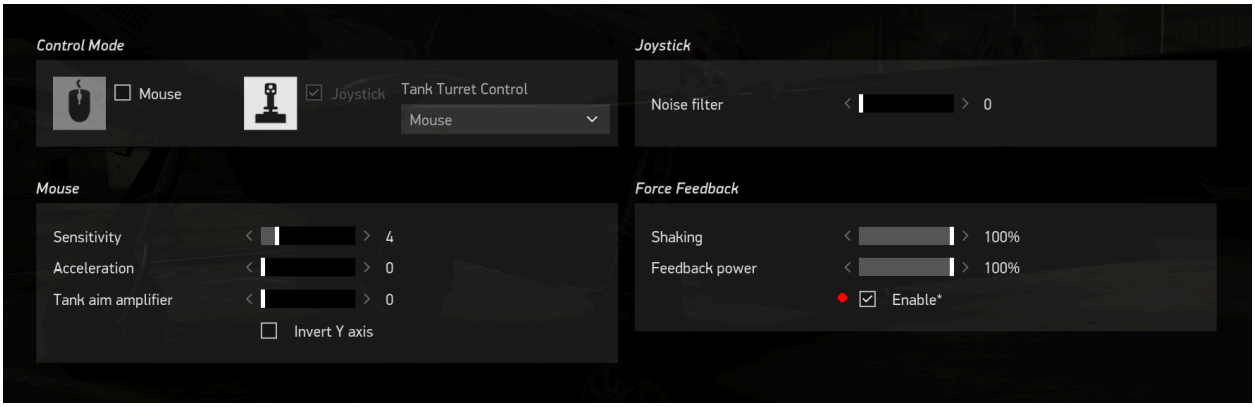




5.2. Il-2 Sturmovik: Great Battles

The latest iteration of Il-2 supports force feedback directly through DirectInput, which makes it very simple to get working. Supported effects include control stiffening with air pressure, stall shake, ground bouncing and gun recoil. Note that the effect of trimming will be different depending on the plane. The level of support is fairly consistent throughout, although thus far only WW1 planes seem to model elevator droop on the ground.

To enable force feedback, simply go to Settings - Input devices and enable Force Feedback as in the image below:



The feedback power setting will adjust the overall spring forces applied by any given aircraft. The ‘shaking’ setting will affect the haptic feedback effects like gunfire and stall buffeting.

Some glitches such as force reversion can occur. Sometimes the simulator rights itself simply after a restart. If reversion occurs after reversing axis in game, the nuclear option is to do the reversal in FFB Configurator software. This will of course affect every other software that has something to do with the RHINO.

Other glitches may occur when changing windows focus to another window and then back to IL-2. Depending on the aircraft, you may notice loss of spring forces in one or both axes, or a large shift in the center point of the spring forces.

5.3. Microsoft Flight Simulator

The latest iteration in the MSFS franchise does not offer native support for force feedback. It does offer a multitude of exports, though, which enables external software such as XPForce to approximate force feedback effects from the telemetry.

The TelemFFB application implements full support for FFB with MSFS 2020.

5.4. X-Plane

The X-Plane franchise does not offer native support for force feedback. It does however have a full SDK that can be used to access the necessary telemetry data to implement FFB externally. The TelemFFB application enables full support for FFB with X-Plane 11/12 by implementing its own plugin which is installed into X-Plane that enables exporting telemetry data.

5.5. IL-2 Cliffs of Dover

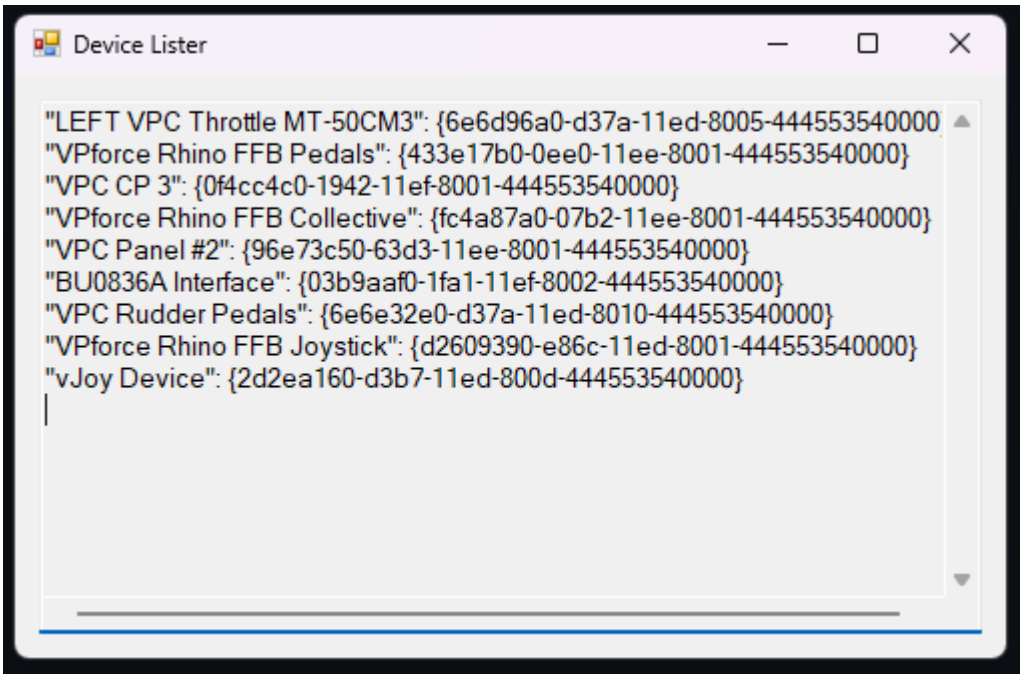
CLoD supports DirectX FFB but has issues with device enumeration. As soon as it finds a joystick device that advertises FFB support, that is where it stops looking. Any other FFB device that is connected is ignored in terms of FFB generation.

For many people this first device may be 'vJoy Device', even if FFB is disabled in the vJoy settings. The result is that "FFB does not work", when in reality, CLoD is sending FFB commands to vJoy, or, in the case of users with multiple FFB devices (i.e. pedals or collective), to the wrong FFB device.

The solution is to use devreorder to fix the enumeration order of the joystick devices on your system (https://github.com/briankendall/devreorder/releases).

After unzipping the devreorder package:

Run DeviceLister.exe and make note of the output:



This is the exact order in which CLoD will see the joystick devices. You can see from the output that my FFB Pedals appear before my Joystick. The end result is that my Pedals are being given the FFB effects during game play.

Here is the procedure to resolve the issue:

- From the devreorder package, make a copy of **devreorder.ini** and place it in the root folder of CLoD (i.e. *Program Files (x86)\Steam\steamapps\common\IL-2 Sturmovik Cliffs of Dover Blitz*)
- From the devreorder package, copy **x64/dinput8.dll** and place it in the **/parts/core** folder of your CLoD install.
    - **Note**, the instructions for performing this that are hidden away in the game help document make specific reference to the x86 version of the dll file. However it seems the game has been updated to 64bit and requires the x64 version (as of 12/20/24 using standard steam install)
- **Open DeviceLister.exe** from the devreorder package
- **Edit the new devreorder.ini file** that you placed into the game folder
    - Follow the guideline documentation inside the ini file but, in general, paste the **NAMES only** of the devices (excluding the quotes and GUID values) , in the order you want them to be enumerated, into the **[order]** section of the ini file.
    - Make sure that your VPforce Rhino Joystick (or appropriate name) is listed before any other FFB capable device (including vJoy!!!)

Using the above devicelister output as an example, here is the resulting ini file [order] section:

```
[order]
; In this section write the names or GUIDs of your controllers
; in the order you want them to be detected, one per line.
; Make sure any names exactly match the name printed in the
; Game Controllers control panel or in DeviceLister.exe,
; including any capital letters and punctuation. If you use
; a device's GUID instead of its name, make sure the GUID is
; enclosed in curly braces, just like in DeviceLister.exe.
; Example:
; vJoy Device
; Controller (XBOX 360 For Windows)
VPforce Rhino FFB Joystick
LEFT VPC Throttle MT-50CM3
VPforce Rhino FFB Pedals
VPC CP 3
VPforce Rhino FFB Collective
VPC Panel #2
BU0836A Interface
VPC Rudder Pedals
vJoy Device


[hidden]
```

You can see here that the only difference is that I have placed my Joystick at the top of the list.

After making these changes, the FFB effects are applied to the correct device.

Note: I have read many reports of general FFB issues with CLoD, often resulting in people modifying the “ffe” files that are included with the game. That is outside the scope of this guide.

This guide is only intended to get CLoD to use the correct FFB device

5.6. Prepar3D

Left blank for future TBD

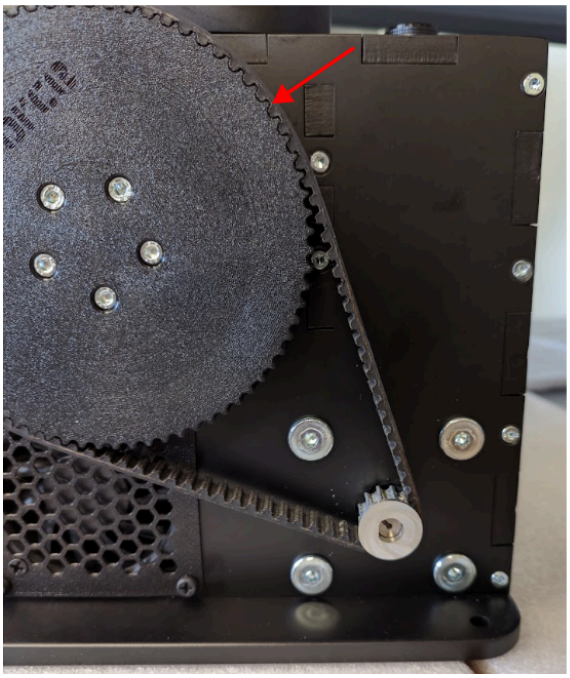
6. Troubleshooting & Maintenance

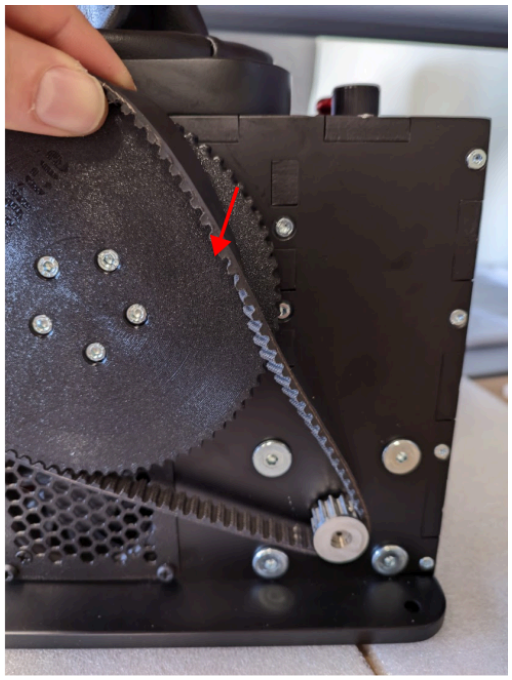
6.1. Re-tightening the belts

The belts might need re-tightening during the device's lifetime. The tightening procedure is quite straightforward. The holes that motor bolts are held are slotted so the procedure is done by pushing the motor back slightly with the belt slipped off.

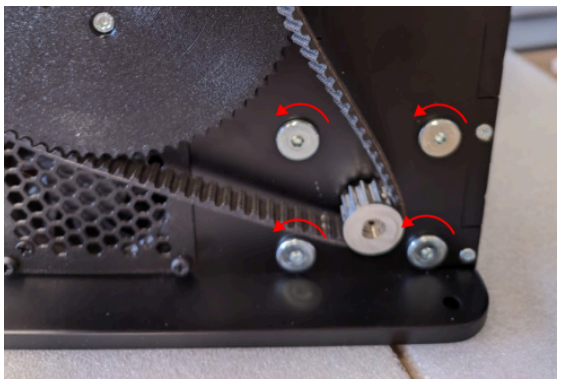
Note: the roll (X axis) motor on the *Rhino* is slotted vertically so it will need to be pushed down.

Step 1: Slide off the belt like shown in the image, but to maintain the existing calibration, do not completely remove it.

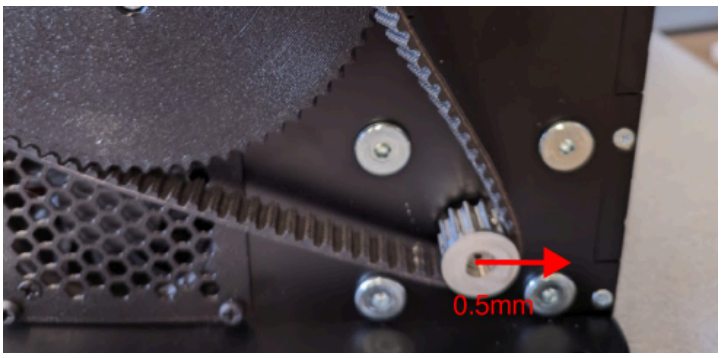




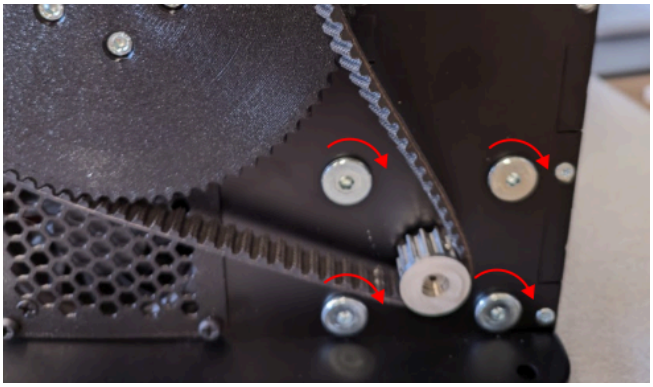
Step 2: Loosen the motor screws. This will allow the motor to slide in the slots.



Step 3: Push the motor back ~0.5mm



Step 4: Re-Tighten the bolts (or a single bolt to test the tension)

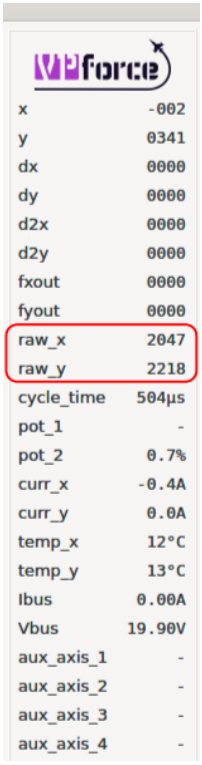


Once the motor bolts are tightened, you can slide the belt back on. If the tension is too high, indicated by the inability to slip the belt back on the large gear, you can loosen the bolts again and slide the motor slightly towards the large gear and repeat step 4.

Note: If the belt slipped, the motor pulley / stick pulley relationship needs to be realigned before reattaching the belt.

To do this, turn on the Rhino and turn the motor pulley so that raw_x or raw_y value, depending on the axis in question, is close to 2100. Then while the stick is roughly centered, reattach the belt.

After the procedure the Auto Calibration will need to be performed again, ideally the calibrated values will stay in the 0 .. 4096 range.



6.2. Stick drifts when trimming off center

See section 3.4 Balancing the Grip

6.3 Clicking Noise During Force Reversal

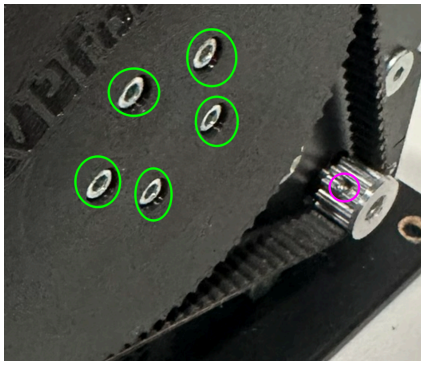
If you notice a clicking sound or slight looseness when the force direction reverses, it may be due to mechanical play or backlash in the VPforce Rhino hardware. This is often caused by slight movement between components when force is applied in opposite directions. Common reasons include slightly loose joints.

Below are typical sources of backlash, listed in order of how easy they are to check and adjust:

1. **Small 12T Pulley on the Motor Shaft**

   Most common cause of noise due to micro-slippage on the shaft.

   This pulley is secured by three M3 grub screws (2 mm hex) spaced 120° apart. If these are not tightened firmly, the pulley may slip slightly under load changes, causing a click or knock.

   - Use a high-quality 2 mm hex key to avoid stripping the screw heads.
   - Apply firm torque and ensure all three screws are tightened evenly.

2. **HEX Bolts on the Large Gears**

   Rare source of noise, but very easy to check.

   Each large gear (on both axes) is fastened with five M4 HEX bolts.

   - Simply check and tighten these bolts. Even if they are not the root cause, this step takes little effort and helps rule them out.

3. **Axis Coupling in the Center of the Gimbal (X and Y Axes)**

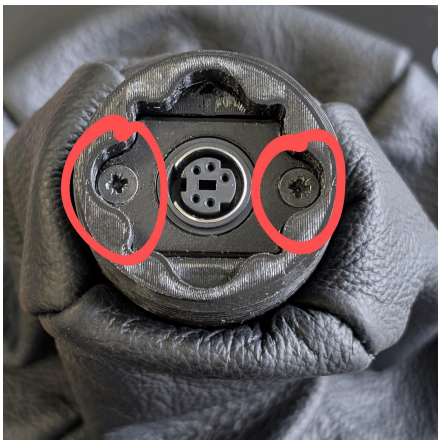
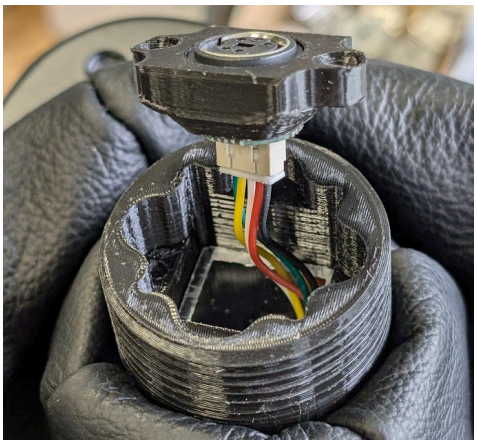
   A slightly loose coupling here can cause a "clunk" sound during reversal.

   The X and Y axes are joined via a central bolt that passes through the gimbal and clamps the axes via bearings. If this bolt/nut assembly becomes even slightly loose, the nut may shift slightly under force over the bearing, producing a "clunk".

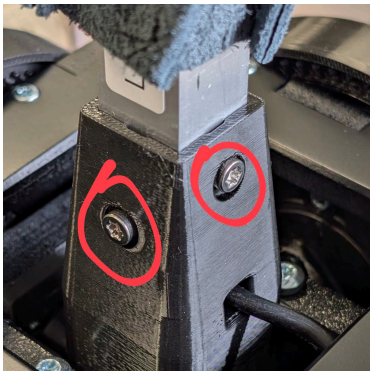
Note: For the **Y axis**, everything is the same except this axis coupling nut does **not** affect it — force is transferred directly into the gimbal stem, bypassing this joint.

How to access and tighten (relevant to X axis):

1. **Lift the leather cover** — do not cut the zip tie.
2. **Remove the grip connector** by unscrewing the two *PH1* screws, pulling out and disconnecting the grip connector. Insert a tool into a screw hole to aid in pulling the socket out. Simultaneously, push the cable under the leather into the stem. This will help lift the socket out.

3. **Detach the gimbal stem**:
    - On older Rhino units: remove the **four** *PH2* screws from all sides.
    - On newer units: remove the **four** *TX20* screws from all sides.
    - Pull out the aluminum tube.



4. Inside the center of the gimbal you'll find an **M5 nut**. Tighten this nut securely using a socket driver from the top.
    - In most cases, tightening from the top is enough.
    - If the bolt spins freely (i.e., the nut doesn't tighten), access the **bottom of the gimbal**:
        - Remove the bottom plate of the unit.

- Hold the PH2 bolt head in place with a screwdriver while tightening the nut from the top.

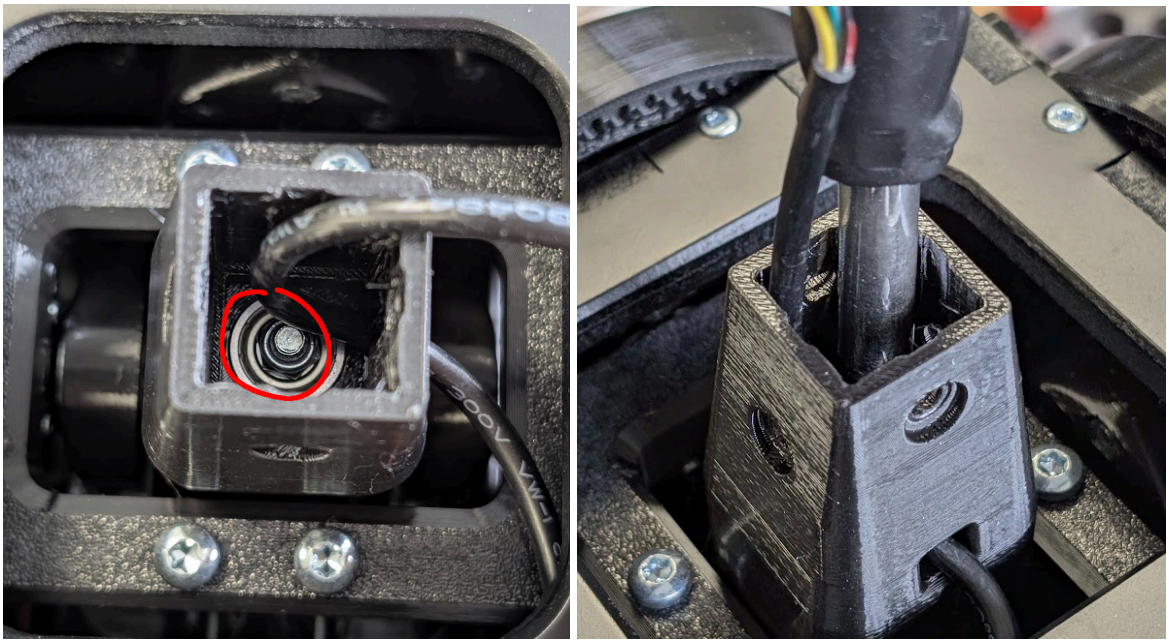


Figure: Tightening the gimbal center nut

6.4 Replacing the Grip Mount Head

Materials Needed:

- TX20 driver/bit or PH2 (for older units)
- TX10 bit
- PH1 driver/bit
- 1 pc zip tie

Instructions:

1. **Remove Dust Boot:**
    - Remove the 4pcs T10 screws that secure the dust shoe to the base.
    - Cut the zip tie that holds the leather in place and then remove the dust boot.
2. **Disconnect Grip Connector Socket:**
    - Remove the 2 PH1 screws that secure the grip connector socket.
    - Disconnect and remove the socket.
3. **Remove Old Grip Mount:**
    - Remove the 4 T20/PH2 screws holding the grip mount.
    - Take off the old grip mount, making sure to note the orientation of the star pattern for proper reinstallation.
4. **Install New Grip Mount:**
    - Place the aluminum grip mount in the correct orientation.
    - Reassemble by screwing all components back in the reverse order of removal.

7. Appendix A: Known issues

The following is a collection of issues that are known to cause issues with the VPforce Rhino or FFB in general:

If you have other issues you feel would be useful to document here, please post in the discord.

DCS

1. Axis curves are generally incompatible with FFB. This also applies to axis saturation settings in DCS. When curves or axis saturation are used, the FFB absolute position is no longer in sync with the logical position as a result of the curve. This will result in incorrect trimming, autopilot issues and particularly bad force-trimming in helicopters
2. The SimHaptic application by rkApps has an “auto start” feature that will launch the application when DCS starts. It is unknown whether the issue is on the SimHaptic side or the DCS side, however this auto-start feature will interfere with and stop FFB from working in DCS. Disabling the auto-start feature in this app will resolve the issue (Issue still exists as of April 2025)
3. vJoy can also cause issues with FFB device detection in DCS. Ultimately, DCS only supports 1 FFB device. If you are experiencing issues with FFB properly working in DCS, check the DCS logs for vJoy starting and see if it indicates support for FFB in the log message. Either uninstall vJoy or disable the FFB capabilities of vJoy to resolve

IL2

1. IL-2 only supports 8 USB devices. It is common for users to have more than 8 peripheral devices. While possible to unplug un-needed devices during gameplay so that IL2 will see your Rhino devices, a better solution is to use devreorder to alter which devices IL2 will see and in what order they will appear.

8. Appendix B: Legacy TelemFFB Documentation

8.1. Installing the TelemFFB executable version

TelemFFB is a python application, however it is also released as a packaged executable that can be run without installing any additional software on your computer.

Downloading the Main Branch release:

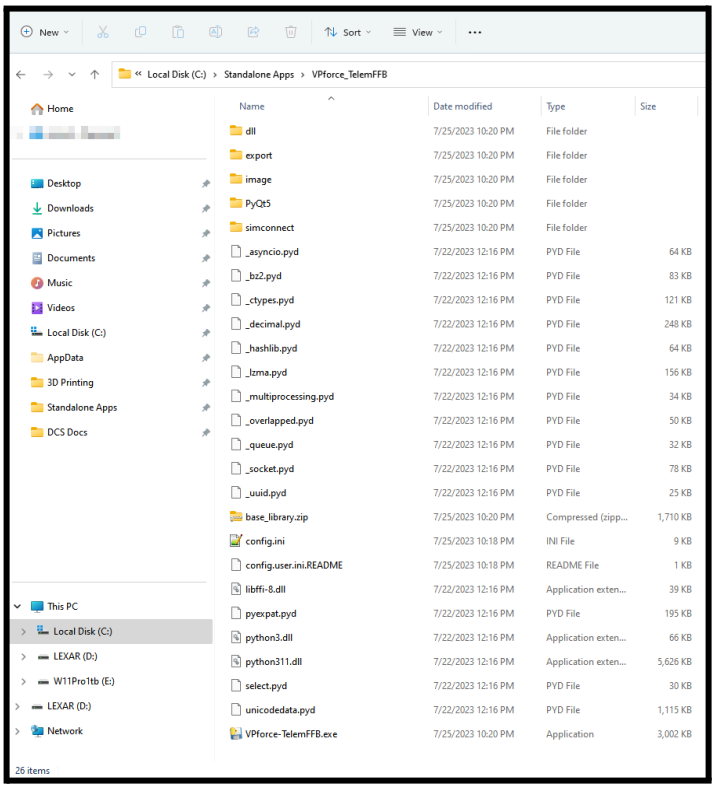
The latest main branch executable release can be found on the releases page on GitHub.

Downloading the WIP Branch release:

The WIP branch undergoes an auto-build process after each commit of updated code to the repository. The latest WIP executable builds can be found on the VPforce Controls website.

Installation:

Once you have downloaded the release zip file, simply extract the archive on your computer. The location itself does not matter.



8.2. Running the TelemFFB Application

To start the application, double-click “VPforce-TelemFFB.exe”

If you are using the python source code, start the application by running “python main.py”

8.2.1. Application Overrides

There are a variety of command line arguments that can be used at runtime:

- `-D | --device <vid:pid>`
  - Specify the VID:PID of the VPforce device. Default FFFF:2055
- `-r | --reset`
  - Reset the VPforce device and clean up any lingering effects. Note: Is destructive to any active effects being generated by a simulator.
- `-c | --configfile`
  - Override the primary configuration file that is loaded. Default: ‘config.ini’
  - **Supported, but recommended to use override option below**
- `-o | --overridefile`
  - Override the user override file to be loaded. Default: ‘config.user.ini’
- `-s | --sim <DCS|MSFS|IL-2>`
  - Enable the defined simulator, regardless of setting in config file. Default ‘DCS’
- `-t | --type <joystick|pedals>`
  - Tell TelemFFB that the device being connected to is a joystick base or rudder pedals (certain effects only apply to one or the other, or behave differently). Default ‘joystick’

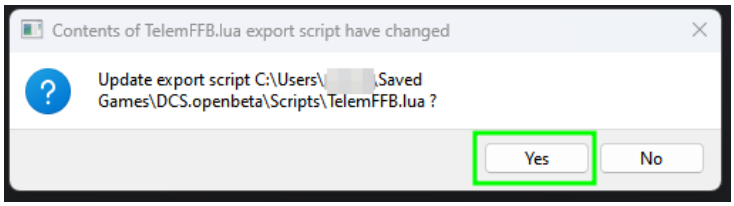
You can permanently enable or disable a given simulator inside of your primary or user override configuration file. Keep in mind that an overlapping setting in the override file will supersede the setting in the primary config file.

```
[system]
logging_level = INFO                    # DEBUG | INFO | WARNING | ERROR | CRITICAL
telemetry_timeout = 200ms     #ms - increase only if low frame-rate resulting in lost effects/control forces due to timeout
msfs_enabled = no                        # yes/no 0/1 | can also run app with '-s MSFS' argument
dcs_enabled = yes                        # yes/no 0/1 | can also run app with '-s DCS' argument
il2_enabled = no                        # yes/no 0/1 | can also run app with '-s IL2' argument
il2_telem_port = 34385                                          # port number for Il-2 telemetry senders
il2_path = 'C:\Program Files\IL-2 Sturmovik Great Battles'          # Path to IL2 root directory
```

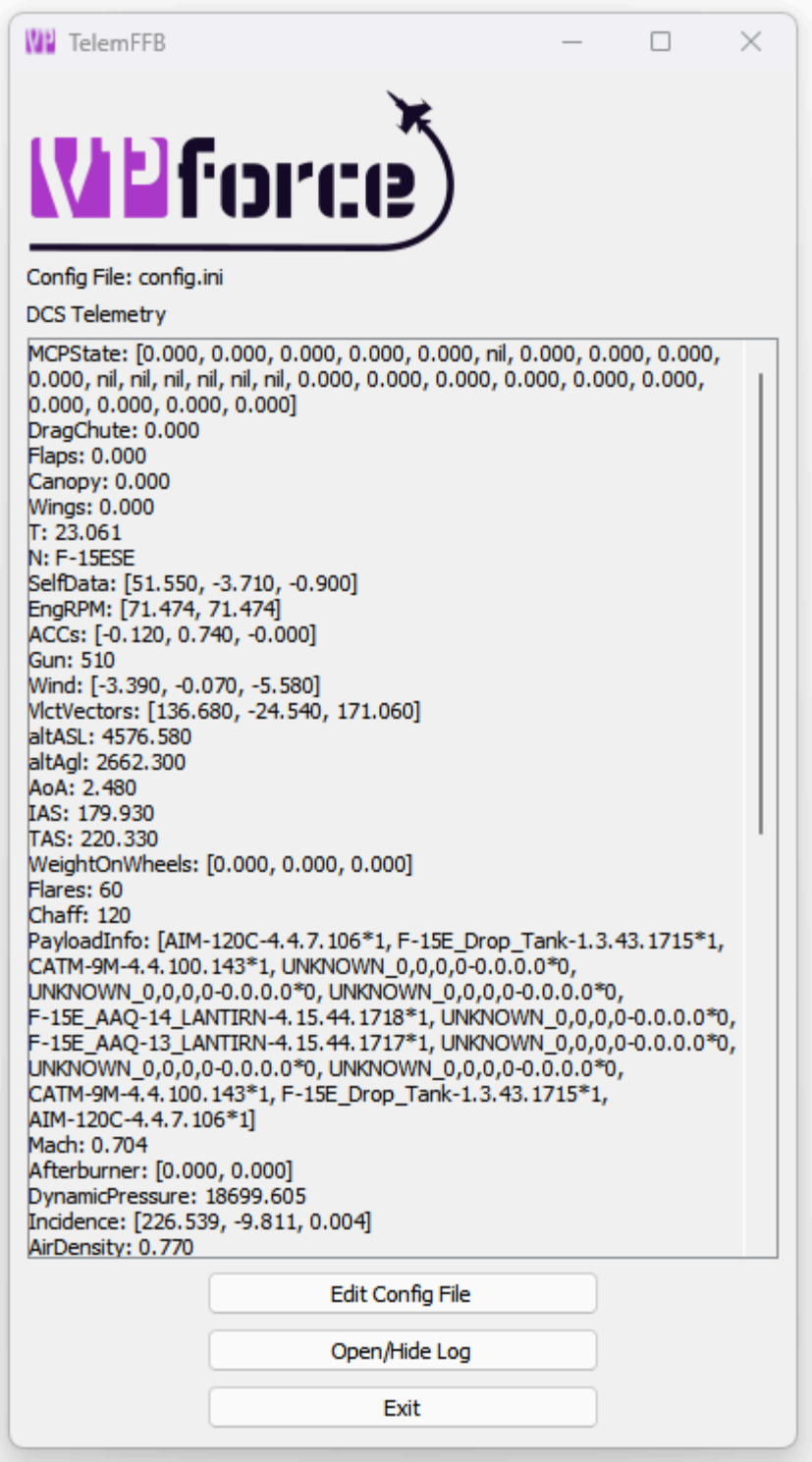
8.3. Setting up TelemFFB for DCS

DCS uses a telemetry export LUA file to define the data and parameters that need to be sent to TelemFFB. The setup of this file and the modification of the export.lua file in your DCS 'saved games' folder is handled automatically by TelemFFB.

The first time you launch the application, or if a new version of TelemFFB needs to update the DCS export file, you will receive a popup notification asking if you want to update. Selecting 'Yes' will update/install the necessary export script file in your DCS saved games folder. The application will then start and wait to receive telemetry from DCS.



Once a DCS module is loaded , the TelemFFB window will update with real-time telemetry statistics. This is your indication that everything is working properly.


TelemFFB
Config File: config.ini
DCS Telemetry
MCPState: [0.000, 0.000, 0.000, 0.000, 0.000, nil, 0.000, 0.000, 0.000, 0.000, nil, nil, nil, nil, nil, nil, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000]
DragChute: 0.000
Flaps: 0.000
Canopy: 0.000
Wings: 0.000
T: 23.061
N: F-15ESE
SelfData: [51.550, -3.710, -0.900]
EngRPM: [71.474, 71.474]
ACCs: [-0.120, 0.740, -0.000]
Gun: 510
Wind: [-3.390, -0.070, -5.580]
VlctVectors: [136.680, -24.540, 171.060]
altASL: 4576.580
altAgl: 2662.300
AoA: 2.480
IAS: 179.930
TAS: 220.330
WeightOnWheels: [0.000, 0.000, 0.000]
Flares: 60
Chaff: 120
PayloadInfo: [AIM-120C-4.4.7.106*1, F-15E_Drop_Tank-1.3.43.1715*1, CATM-9M-4.4.100.143*1, UNKNOWN_0,0,0,0-0.0.0.0*0, UNKNOWN_0,0,0,0-0.0.0.0*0, UNKNOWN_0,0,0,0-0.0.0.0*0, F-15E_AAQ-14_LANTIRN-4.15.44.1718*1, UNKNOWN_0,0,0,0-0.0.0.0*0, F-15E_AAQ-13_LANTIRN-4.15.44.1717*1, UNKNOWN_0,0,0,0-0.0.0.0*0, UNKNOWN_0,0,0,0-0.0.0.0*0, UNKNOWN_0,0,0,0-0.0.0.0*0, CATM-9M-4.4.100.143*1, F-15E_Drop_Tank-1.3.43.1715*1, AIM-120C-4.4.7.106*1]
Mach: 0.704
Afterburner: [0.000, 0.000]
DynamicPressure: 18699.605
Incidence: [226.539, -9.811, 0.004]
AirDensity: 0.770
Edit Config File
Open/Hide Log
Exit

8.3.1. Notable DCS Specific Settings

The following sections document some of the 'non haptic effect' features of TelemFFB for DCS.

8.3.1.1. Trimming for VPforce powered DIY FFB Pedals

As described in 5.1.1 - Setup tips for DIY FFB Pedals and/or Collective devices, DCS does not properly support FFB pedals. As such, the following implementation has been added to TelemFFB to enable both correctly behaving spring forces as well as trimming for fixed wing aircraft that have rudder trimmers.

Helicopter trimming is not currently supported as there are currently no viable methods to deal with the "double input" effect that is generated by the "instant trim" option for those helicopters which support pedal trimming. Additionally, helicopters like the Mi-24 implement an approximation of the real helicopter's "foot microswitch" logic which detects when the pilot's feet are on the pedals. None of the modes for this simulation of that switch logic are conducive to integrating with FFB trim following.
The shining light is that with the auto-switching to springless mode for helicopters, pedal trimming is not really necessary.

The following settings make up the feature in the TelemFFB configuration:

- `pedal_trimming_enabled - (True/False, default=True)`
    - Enabled by default for both JetAircraft and PropellerAircraft types

8.3.1.2. Auto pedal spring mode switching duplicated

In addition to pedal trimming, TelemFFB now implements dynamic switching between 3 different modes for Helicopters (mode 1), Jets (mode 2) and Prop aircraft (mode 3). The modes may be overridden on a per aircraft basis by adding the applicable mode setting to that aircraft section in the configuration.

All of the DCS warbirds have default values built into the application for the V speeds. It is possible to override the default internal V_S and V_{NE} speeds as well as the spring gains.

The settings which make up this feature are as follows

- `pedal_spring_gain (0-100%)`
    - Percent of max value set in VPforce configurator
- `pedal_spring_mode`
    - 0=DCS Default
    - 1=spring disabled
    - 2=static spring enabled using "pedal_spring_gain" spring setting
    - 3=dynamic spring enabled. Based on "pedal_spring_gain", dynamic force between 0 and Vs speed (%25 of force) and Vs and Vne speeds (remaining %75)
- `pedal_dampening_gain (0-100%)`
    - Percent of max value set in VPforce Configurator

To change the V speeds or add V speeds to a non-warbird type aircraft, or adjust the gain values, you can add the following settings to your aircrafts configuration

- `pedal_spring_mode = 3` (to enable dynamic mode)
- `aircraft_vs_speed` - Stall speed in m/s
- `aircraft_vs_gain` - Spring gain when aircraft at Vs (0-100%)
- `aircraft_vne_speed` - Never Exceed speed in m/s
- `aircraft_vne_gain` – Spring gain when aircraft is at Vne (0-100%)

8.4. Setting up TelemFFB for MSFS 2020

Microsoft Flight Simulator does not require any special configuration on the simulator side. All subscriptions to telemetry data are done via the SimConnect API.

Inside of the 'config.ini' configuration file, near the top inside the `[system]` section is a `msfs_enabled` flag. Setting this flag to 'yes' or '1' will enable the code which attempts to connect to, and listens for, simconnect events from MSFS.

You may also enable MSFS communication when starting TelemFFB from the command line using the '-s' parameter (i.e., `-s MSFS`)

You will know you have successfully enabled support for MSFS when you see the corresponding indication in the TelemFFB window:

As with DCS, when you load into a flight (or when MSFS first enters the hangar/menu) you will see the telemetry window populate with values.

It bears repeating here that MSFS does not have native support for FFB devices. Therefore, external software must take in telemetry from the simulator in order to create those basic dynamic spring forces impacted by prop wash and airspeed as well as other common FFB type effects like ground-roll/touchdown and stall buffeting.

For DCS (actually it is implemented per aircraft module by the individual developers) the game itself supplies these basic effects to FFB Joysticks.

8.4.1. Notable MSFS Specific Settings

The following sections document some of the 'non haptic effect' features of TelemFFB for MSFS.

8.4.1.1. Axis Position Sending via SimConnect (duplicated in settings section

The trim/ap following feature requires that TelemFFB be the source of the axis position for MSFS. As such, the following settings will enable and configure the sending of the axis positions via simconnect.

****Note** - You must un-bind your axes in MSFS for this feature to work**

- `telemffb_controls_axes`
  - Master control for the feature (enable/disable, true/false, on/off)
- `joystick_x_axis_scale`
  - Scaling of the joystick X axis (0-100%)
- `joystick_y_axis_scale`
  - Scaling of the joystick Y axis (0-100%)
- `rudder_x_axis_scale`
  - Scaling of the rudder X axis (0-100%)

The input range used by MSFS is -16383 to +16384. Axis curves are not supported in this implementation, however you can utilize the scaling settings to adjust the sensitivity of the physical axis. A (unreasonable) scale value of %50 would send a range of -/+8192 over the full range of the physical axis, resulting in less sensitive control inputs at the expense of range of movement.

The following simconnect events are used to send the axis position data:

Fixed Wing:

- AXIS_AILERONS_SET
- AXIS_ELEVATOR_SET
- AXIS_RUDDER_SET

Helicopter:

- AXIS_CYCLIC_LATERAL_SET
- AXIS_CYCLIC_LONGITUDINAL_SET
- ROTOR_AXIS_TAIL_ROTOR_SET

8.4.1.2. Trim and Autopilot Following

The most common feature request for MSFS is trim and/or AP following. Given the lack of key FFB concepts inside MSFS, such as axis offset, this presents several challenges with such an implementation.

For this reason, this feature as implemented requires TelemFFB to be the source of the axis position data for MSFS (see **Axis Position Sending via SimConnect** above).

See “Tips on configuring the trim settings” at the bottom of this section for guidance on customizing these options per aircraft

**** Hat based trim following is now supported for Generic helicopter types ****

****Autopilot following is only available for fixed-wing aircraft at this time****

**** Autopilot following is part of the trim following feature ****

- trim_following
  - Master control for the feature (enable/disable, true/false, on/off)
- ap_following (fixed wing only)
  - Enable or disable AP following (only disables the aileron/rudder axis). To disable elevator following, you must disable the trim following feature. (enable/disable, true/false, on/off)
- invert_ap_x_axis (fixed wing only)

    - Some aircraft, like the default C172, have a strange behavior where axis inputs, when the AP is active, act in the opposite direction. Note that while this setting allows the aircraft to fly normally, the stick following in the x axis will be incorrect (reversed)
- `joystick_trim_follow_gain_physical_x`
    - Physical stick x axis movement as a percentage of the trim value (0-100%)
- `joystick_trim_follow_gain_physical_y`
    - Physical stick y axis movement as a percentage of the trim value (0-100%)
- `joystick_trim_follow_gain_virtual_x`
    - Virtual stick x axis movement as a percentage of the physical stick movement (0-100%)
- `joystick_trim_follow_gain_virtual_y`
    - Virtual stick y axis movement as a percentage of the physical stick movement (0-100%)
- `rudder_trim_follow_gain_physical_x (fixed wing only)`
    - Physical pedal x axis movement as a percentage of the trim value (0-100%)
- `rudder_trim_follow_gain_virtual_x (fixed wing only)`
    - Virtual pedal x axis movement as a percentage of the physical pedal movement (0-100%)

How it works at a high level:

- Trim Following
    - Trim position is read from the sim
    - Physical stick center point is calculated using the ‘physical’ position gain
    - Physical stick center is sent to the joystick/pedals
    - Virtual stick position is calculated using the ‘virtual’ position gain
    - Virtual stick position is sent to MSFS
- AP Following
    - Elevator AP following is reliant on the trim value, as APs use the elevator trim
    - Aileron/Rudder

        - Control surface deflection is read from the sim (as induced by AP control)
        - Control surface deflection is used to calculate physical axis position
        - Physical position is sent to joystick/rudder
    - The AP induced physical control inputs are dampened to prevent out of control oscillations in turbulence or in aircraft with extra sensitive controls.

Tips on configuring the trim settings

Physical & Virtual configuration should be done for each plane.

Suggested starting points:

```
joystick_trim_follow_gain_physical_x = 50%
joystick_trim_follow_gain_virtual_x = 20%

joystick_trim_follow_gain_physical_y = 100%
joystick_trim_follow_gain_virtual_y = 20%

rudder_trim_follow_gain_physical_x = 50%
rudder_trim_follow_gain_virtual_x = 20%
```

Joystick..X and Rudder..X can typically be left as default, since many planes do not even have in-cockpit trims on those axes, and if they do they are set and forget. The elevator trim however is interacted with a great deal and joystick..Y must be tuned per plane for realistic results.

Create a custom plane entry in the config.user.ini. There are some default aircraft included as examples in config.ini. Save and restart TelemFFB.

Adjust joystick_trim_follow_gain_physical_y % to the amount that trim can move the surface vs the full travel of the elevator. If it's possible for full trim to fully deflect the elevator, this will be 100%.

Then, fly the plane, and trim for level flight at cruise speed.

In VPForce configurator, temporarily set spring to 0% and friction to 100%. Apply (do not store) the setting.

Without moving the Rhino joystick, use your trim buttons/keys/axis to nose down the plane.

If the nose goes up, adjust joystick_trim_follow_gain_virtual_y 10% higher.

If the nose goes down, adjust joystick_trim_follow_gain_virtual_y 10% lower. It may be negative.

Save the config.user.ini and the effects will be immediate.

Adjust the trim and observe the reaction again. It will take a few iterations. The goal is to have the trim adjustment have no effect with the stick not moving. You can adjust by 5%, 1% when you are close.

When finished, go back to VPForce configurator and click Load Settings, your Friction and Spring will return to your stored settings. Enjoy your new realistic trim!

8.4.1.3. Helicopter Force Trim

TelemFFB implements a hardware based force trim feature for MSFS helicopters. The implementation is identical to how hardware trim works when configured inside VPforce Configurator, however the benefit of doing it inside TelemFFB is that it is dynamically enabled when loading a helicopter. Enabling this in VPforce Configurator requires enabling/disabling 'sticky spring' and force trim when switching between helicopter and fixed wing types.

- `force_trim_enabled`
  - Master control (enable/disable, on/off, true/false)
- `cyclic_spring_gain`
  - Percent of VPforce Configurator spring value (0-100%)
- `force_trim_button`
  - joystick button to use for force trim (button number as seen by vpforce configurator)
- `force_trim_reset_button`
  - joystick button to use for trim reset (button number as seen by vpforce configurator)

8.4.1.4. Glider Force Trim

Many gliders have a lever actuated trim positioning system that recenters the elevator trim to hold the control stick where the lever is released. This is modeled in TelemFFB per the below settings

- `force_trim_enabled`
  - Master control (enable/disable, on/off, true/false)
- `elevator_force_trim`
  - Enable/disable force trim functionality on the elevator axis (true/false, on/off, enable/disable)
- `elevator_force_trim`

    - Enable/disable force trim functionality on the elevator axis (true/false, on/off, enable/disable)
- `force_trim_button`
    - joystick button to use for force trim (button number as seen by vpforce configurator)
- `force_trim_reset_button`
    - joystick button to use for trim reset (button number as seen by vpforce configurator)

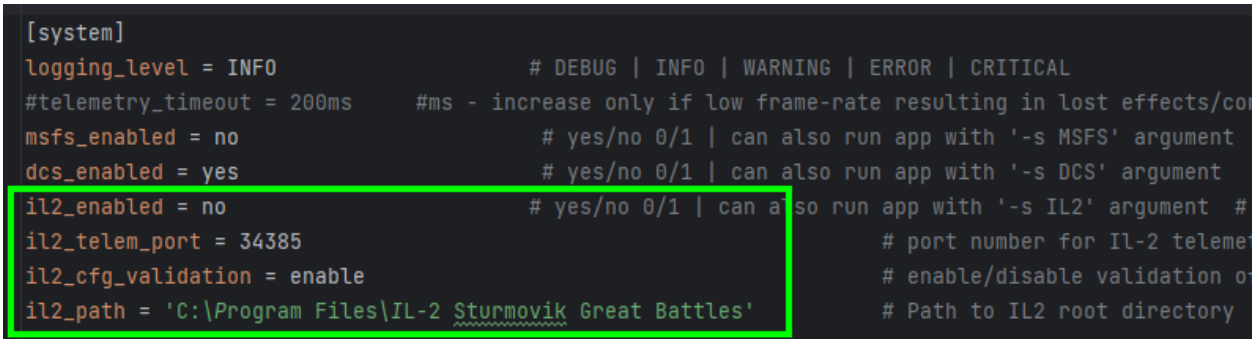
8.5. Setting up TelemFFB for IL-2 Great Battles

Inside of the 'config.ini' configuration file, near the top inside the `[system]` section is an `il2_enabled` flag. Setting this flag to 'yes' or '1' will enable the code which listens for telemetry from IL-2.

You may also enable IL-2 communication when starting TelemFFB from the command line using the '-s' parameter (i.e., `-s IL2`)

IL-2 Great Battles requires that the receiver address and port values be configured inside of the `'/data/startup.cfg'` file in the simulator installation directory.

TelemFFB can automatically add the required entries to the file provided the following information is supplied in the TelemFFB configuration file.

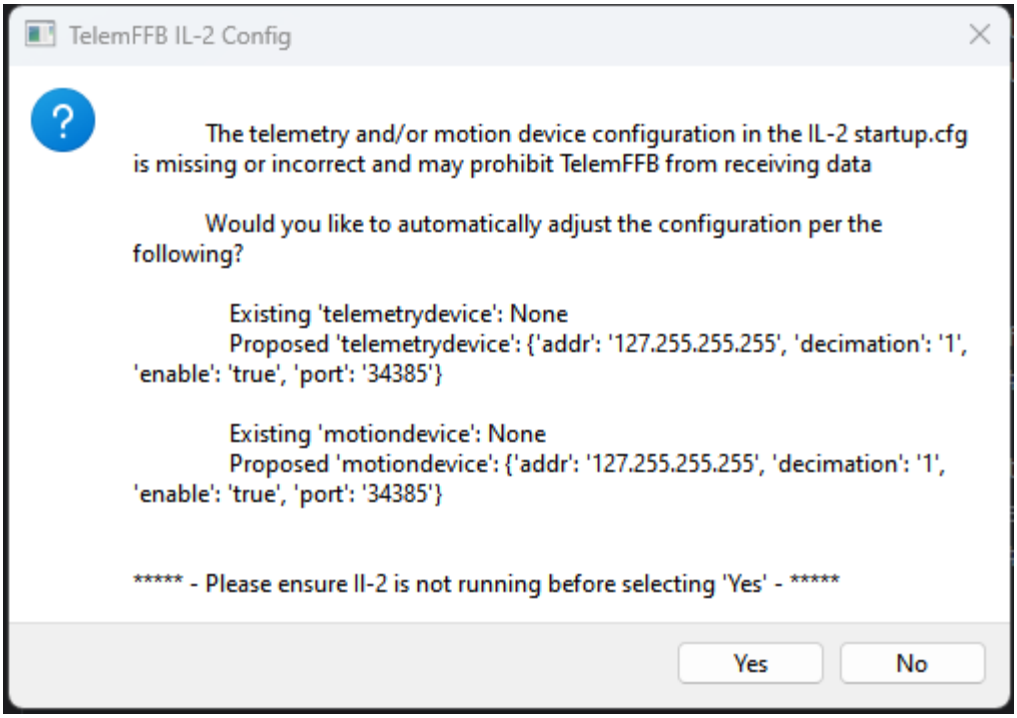


`il2_telem_port` - This is the port number that will be configured for the telemetry receiver in the IL-2 config file

`il2_cfg_validation` - If you want to manually configure your IL-2 telemetry settings, set this to 'disable'

`il2_path` - This is the root directory of your IL-2 installation.

The first time you launch TelemFFB with IL2 enabled, **if you do not already have any telemetry receivers defined**, you will see the following notification.



These are the entries that will be placed in startup.cfg which will enable telemetry to be sent to TelemFFB. **Do be certain that IL-2 is not running when you select Yes**. Otherwise, IL-2 will delete the entries from the config when you restart it.

8.5.1. Notable IL-2 specific settings duplicated

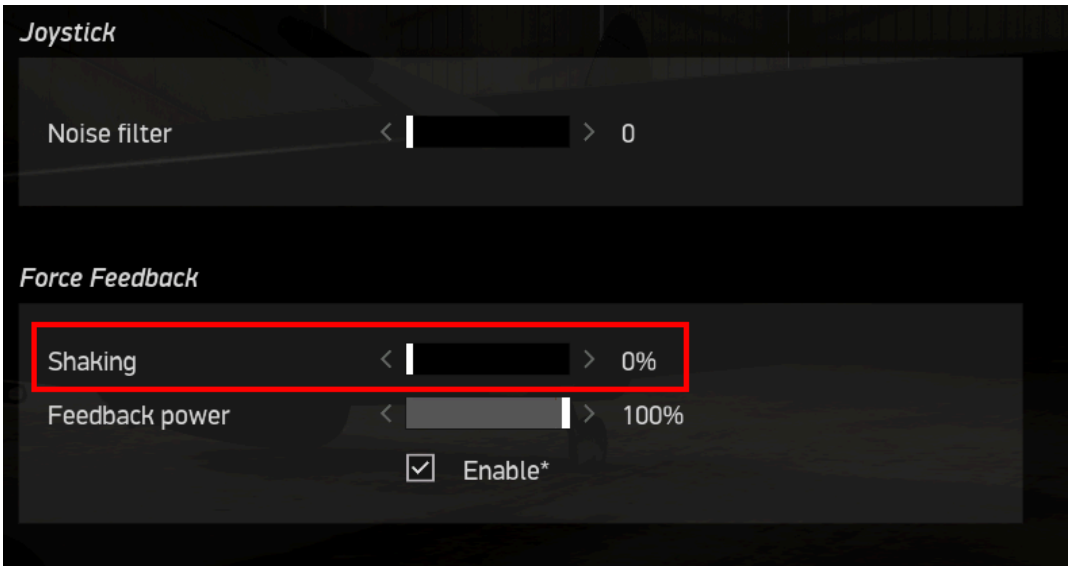
While the majority of the settings for use in IL-2 are similar or identical to those that are used in DCS and MSFS, there are several that differ.

IL-2 has several native FFB effects that can overlap with what TelemFFB is capable of generating. Specifically weapons release, runway rumble and buffeting. The benefit of implementing these in TelemFFB is that each effect is individually configurable both from an enable/disable perspective as well as the intensity.

In the configuration, there is a section that is controlled by a master on/off variable:

```
il2_shake_master = off   ## master control for effects which
    il2_enable_weapons = yes
        weapon_release_intensity = %20
    il2_enable_runway_rumble = yes
        runway_rumble_intensity = 10%
    il2_enable_buffet = yes
        il2_buffeting_factor = 1.0
```

These three effects are generated by IL-2 by default. If you wish to use the effects generated by TelemFFB in lieu of those generated by IL-2, you can enable the master setting in the TelemFFB config and then disable the 'shaking' effects in the IL-2 FFB settings as follows. Navigate to Settings->Input Devices and move the 'Shaking' slider to 0.



8.5.2. Coexistence with other telemetry receivers (Base shaker, motion platforms, etc)

IL-2 only supports a single TelemetryDevice and a single MotionDevice configuration. However, the configuration can make use of multiple receiver addresses, provided the additional receivers are configured correctly.

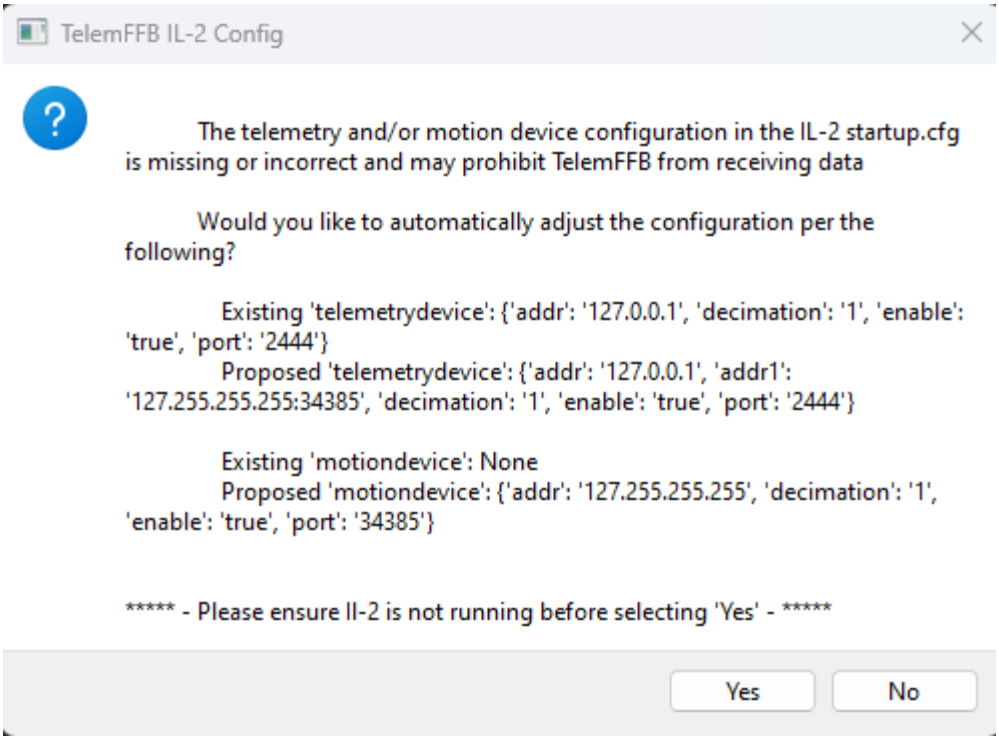
TelemFFB uses the link local broadcast address (127.255.255.255) because if multiple VPforce devices (joystick, pedals, etc) are in use, multiple instances of TelemFFB need to be able to receive the data. Rather than configuring a unique IP or port per instance, telling IL-2 to send to the broadcast address will allow any host listening to that port with a 127.0.0.0/8 address to receive the data.

In the event that a previous configuration exists for either telemetry or motion, TelemFFB will attempt to insert its own entry as an additional receiver within the config.

Note that TelemFFB will attempt to enforce a **decimation** value of 1. This value affects the frequency of the telemetry data. Some apps (like SRS for IL-2) call for a value of 2. This works fine if you are only concerned with server information, but for a real-time telemetry application such as TelemFFB, decimation values higher than 1 may produce unexpected results in the generated effects.

Other applications that may call for a decimation value of 2 should work with a decimation value of 1, even if they don't need the extra data fidelity.

If this is the case, you will see a notification from TelemFFB when starting as follows:



If you select ‘Yes’, the configuration will be overwritten with the proposed values.

8.5.3. Manually configuring the telemetry export for IL-2

While TelemFFB can automatically configure the required entries in the IL-2 startup file, the exact entries that are required are given below in the event you want to manually edit the IL-2 configuration.

Inside of the IL-2 installation folder, navigate to the ‘/data’ directory and edit ‘startup.cfg’.

No existing config:

Insert the following entries (the port numbers may be modified to match what is in the TelemFFB config file (assuming the TelemFFB configuration is specifying port 34385):

```
[KEY = telemetrydevice]
 addr = “127.255.255.255”
 decimation = 1
```

```
 enable = true
 port = 34385
[END]

[KEY = motiondevice]
 addr = “127.255.255.255”
 decimation = 1
 enable = true
 port = 34385
[END]
```

Inserting into existing config:

If there are existing configurations, you need to insert the ‘addr1’ TelemFFB item(s) as follows.

```
[KEY = telemetrydevice]
 addr = “127.0.0.1”
 addr1 = “127.255.255.255:34385”
 decimation = 1
 enable = true
 port = 4222
[END]


[KEY = motiondevice]
 addr = “127.0.0.1”
 addr1 = “127.255.255.255:34385”
 decimation = 1
 enable = true
 port = 4222
[END]
```

8.6. Customizing the TelemFFB configuration

TelemFFB uses a hierarchical configuration model. Within the configuration there are various sections. System settings, Simulator default settings, aircraft type default settings and finally, per-aircraft settings. As the configuration section gets more specific, settings in that section will override the same setting in a higher level section.

Essentially, the “sim default” settings will apply to all aircraft unless they are overridden in an aircraft type or aircraft specific section. The basic layout is as follows:

```
[SIM]
     setting1=value1
[SIM.AircraftCategory1]
     setting1=value2
[SIM.AircraftCategory2]
     setting1=value2
[AircraftName1]
     type=AircraftCategory1
[AircraftName2]
     type=AircraftCategory2
     setting1=value3
```

In the example above, both `[SIM.AircraftCategory1]` and `[SIM.AircraftCategory2]` override the value of `setting1` that is configured at the `[SIM]` level.

The `[AircraftName1]` aircraft will inherit the value of `setting1` from `[SIM.AircraftCategory1]` as it does not further define its own value.

The `[AircraftName2]` aircraft further overrides the value of `setting1` within its own configuration section.

8.6.1. Creating a custom user override file

TelemFFB also uses a config-override model. Inside the main folder structure is a template named 'config.user.ini.README'. If you re-name that file to 'config.user.ini' (remove the '.README'), then any entries in that file will further override the settings which are in the default 'config.ini' file.

This allows users to create and keep their own custom configurations that will not be overwritten when a new version of TelemFFB is installed. In most cases, the user.ini file from one version of TelemFFB will work with a newer version.

While the default config file is 'config.ini', it is possible to use a different config file. While the default config file is generally all that is required for most users, it is useful to use multiple config files if you have multiple VPforce FFB devices (i.e., Rhino and DIY pedals).

Note that any configuration in the user override file must match the hierarchy of the primary configuration file exactly.

TelemFFB will automatically try to load config.user.ini if it exists, however, You can create a user override file with any name you want and pass the file name as an argument to TelemFFB using the "-o <filename>" runtime flag.

8.6.2. Understanding the structure of the config file

The config.ini file is a flat configuration file which consists of multiple sections. At the top are the **system** variables that control certain aspects of the application such as logging level and the flags to enable support for MSFS and/or DCS.

```
[system]
logging_level = INFO                    # DEBUG | INFO | WARNING | ERROR | CRITICAL #MSFS
msfs_enabled = 1                        # 0=disable, 1=enable
dcs_enabled = 0                         # not plumbed - for future use      # DCS
```

System Settings

Below the system settings are the **'default'** category settings. These settings will modify the behavior of their corresponding effects globally for all aircraft unless they are overridden in a specific aircraft block further down in the config (more on that shortly).

Not all settings apply to both DCS and MSFS. Each setting has a comment in-line which indicates the applicable simulators.

```
[default]
type=Aircraft                       # Valid Types 'Aircraft' | 'PropellerAircraft' | 'JetAircraft' | 'Helicopter'
## Default Settings below
## Effects will use default settings below unless un-commented and edited
## Un-comment line to edit, or add to aircraft specific settings below

buffeting_intensity = 0.2           # peak AoA buffeting intensity  0 to disable (default 0.2)                          # DCS/MSFS
buffet_aoa          = 10            # AoA when buffeting starts (default 10)                                           # DCS Only
stall_aoa           = 15            # Stall AoA (default 15)                                                           # DCS Only
runway_rumble_intensity = 0.5       # peak runway intensity, 0 to disable (default 1.0)                                # DCS/MSFS
gun_vibration_intensity = 20%       # peak intenstiy for gunfire effect, 0 to disable (default 0.12)                   # DCS Only
cm_vibration_intensity = 20%        # peak intensity for countermeasure release effect, 0 to disable (default 0.12)    # DCS Only
weapon_release_intensity = 20%      # peak intensity for weapons release effect, 0 to disable (default 0.12)           # DCS Only

......... more settings below ........
```

Default Settings

The final section of the config are the unique aircraft definitions. While the actual order of the aircraft are not important, the default ones included in the config are arranged thusly:

- DCS Aircraft
    - DCS Props
    - DCS Jets
    - DCS Helicopters
- MSFS Aircraft
    - MSFS Props
    - MSFS Turboprops
    - MSFS Jets
    - MSFS Helicopters
    - MSFS Gliders

```
################################################
################################################
########   Aircraft Definitions Below   ##########
################################################
################################################

#########################
#####       DCS     ######
#########################
####################
###  DCS Props   ####
####################
[TF-51D]
type=PropellerAircraft
buffeting_intensity = 0.0 # implemented by DCS
runway_rumble_intensity = 1.0
engine_rumble = 0 # rumble based on RPM
gear_buffet_intensity = 0.15
rpm_scale = 31.5 # RPM% to actual RPM scale

[P-51D]
type=PropellerAircraft
buffeting_intensity = 0.0 # implemented by DCS
```

8.6.3. Customizing per-aircraft settings

Customizing per aircraft settings is fairly straightforward once you understand the basic hierarchy of the config file.

As an example, look at the configuration for the Supermarine Spitfire in DCS:

The most important setting here is the aircraft name "**SpitfireLFMkIX**". This is the unique name in DCS for that aircraft. The next most important field is '**type**'. By setting the type to PropellerAircraft, we are telling TelemFFB that this aircraft is a propeller aircraft and that it should treat it as such and play the effects for that aircraft type based on the values configured in the global settings and/or the individual values in the aircraft config section.

```
[SpitfireLFMkIX]

type=PropellerAircraft

engine_rumble = 1

engine_rumble_lowrpm = 650

engine_rumble_lowrpm_intensity = 0.06

engine_rumble_highrpm = 2800

engine_rumble_highrpm_intensity = 0.03

gear_buffet_intensity = 0.15

buffeting_intensity = 20%

buffet_aoa = 12

stall_aoa = 16
```

If we inspect the config further, we are enabling the engine rumble effect and defining the characteristics of the effect. The 2 RPM and 2 intensity settings work together to define how the effect behaves. At the 'lowrpm' value, the rumble effect will be played at an intensity of %6. As the RPM increases, the intensity will decrease proportionally all the way up to the 'highrpm' value, where the intensity will reach just %3 intensity. Note that these

are not floor values. If the RPM drops below 650, the intensity will increase above %6. Similarly, if the RPM goes above 2800, the intensity will continue to decrease below %3.

Next, we have set the gear buffet effect to %15 intensity. This effect will be played when the gear are extended (proportionally increasing as the gear extend further from stowed to full).

We are also setting the buffeting intensity to %20. Note that you can give the intensity values as either a decimal value 0.00-1.00 or as a percentage 0%-100%.

Lastly we define the buffeting characteristics of the aircraft. The buffeting will begin to occur when the AngleOfAttack reaches 12 degrees. The intensity of the buffetting will start very low and will continue to increase up to %20 at an AoA of 16 degrees. Note that in the case of this particular effect, the intensity value is a ceiling. The intensity will not increase beyond 20%.

All remaining effects that apply the generic and propeller aircraft types will use the default values at the top of the config file

8.6.4. Config files other than ‘config.ini’

While the default ‘config.ini’ will be loaded on startup, it is possible to use a different config file. This can be useful if you are using additional VPforce enabled devices and want or need to have different effects settings for the different devices.

Simply pass the config file name with the ‘-c’ argument. For example:

```
VPforce-TelemFFB.exe -c pedals.ini
```

8.6.5. Auto-loading custom VPForce FFB Configurator profiles

Sometimes it is desirable to have different settings in your VPForce FFB Configurator for different types of aircraft or a specific aircraft. This can be accomplished in TelemFFB

dynamically when the aircraft loads. In your TelemFFB user config file, you can add a “vpconf=” to any sim, aircraft type or specific aircraft section. Normal configuration hierarchy applies.

For example, in the following config excerpt, there are unique vpconf profiles defined at the sim level, the Helicopter category level and a specific aircraft level.

- If the DA-62 is loaded, TelemFFB will call the configurator CLI tool and load the ***da62.vpconf*** profile.
- If any MSFS aircraft that is identified as a helicopter is loaded, it will load ***helicopter.vpconf***
- If any other MSFS aircraft that does not match the two above cases is loaded, it will load the ***default.vpconf*** profile.

Upon successful loading of the profile, you will hear your Rhino beep (as long as the buzzer is not disabled

***** Note that the path is relative to the installation folder where the VPforce_FFB_Configurator executable is.***

```
[MSFS2020]
      type=Aircraft
      vpconf='./configs/default.vpconf'
[MSFS2020.Helicopter]
      vpconf='./configs/helicopter.vpconf'
[DA62.*]
      type=PropellerAircraft
      vpconf='./configs/da62.vpconf'
```

8.7. Effects Documentation (Legacy)

This section attempts to document and explain each of the effects and their settings. It is a work in progress.

A majority of the effects will apply to both DCS and MSFS. Where applicable, each effect setting has a comment in-line with the default setting to indicate which simulator(s) it applies to.

8.7.1. Generic Aircraft Effects (applies to all (or multiple) aircraft types)

8.7.1.1. G-Force Loading Effect

Only applies to fixed-wing aircraft

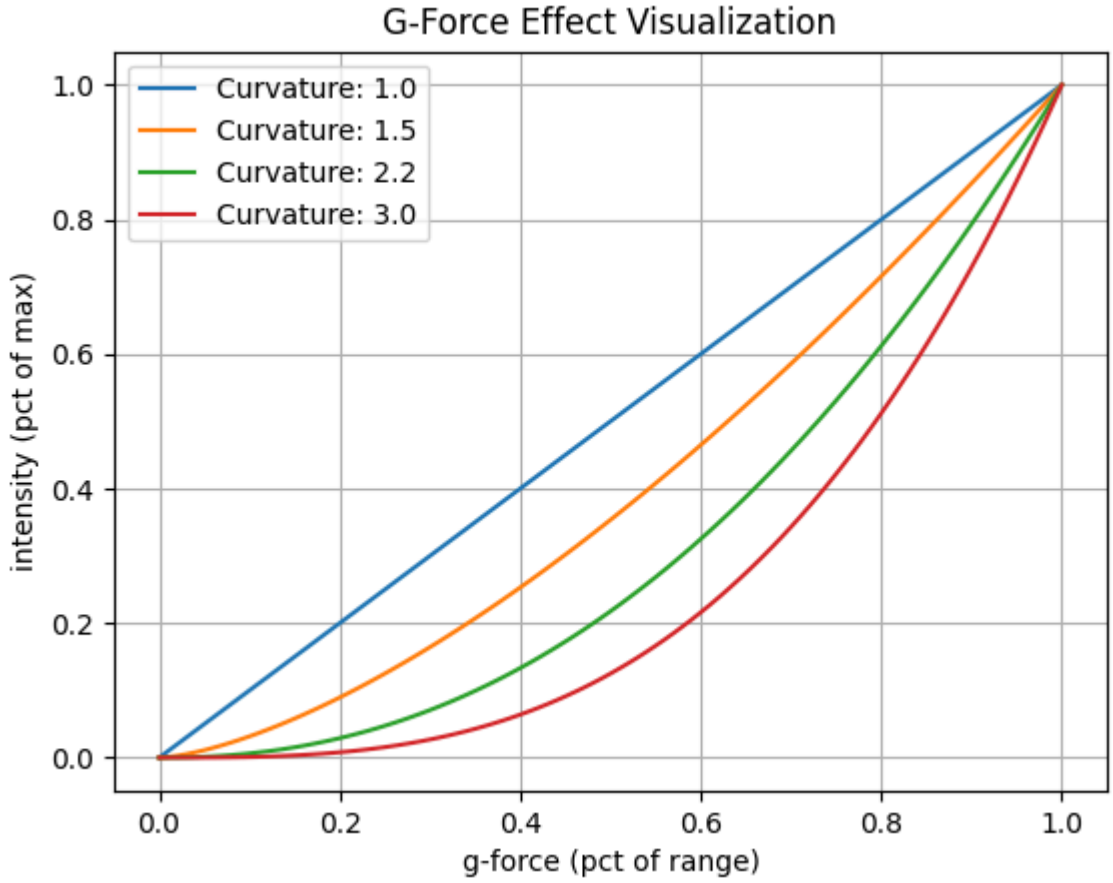
DCS Only (MSFS has built in G forces in the TelemFFB stick force implementation)

The G-Force loading effect simulates the increasing force that is required to pull back on the stick as the g forces increase during a dive pull-out or hard turn.

```
## G force effect will increase the spring force against movement as the G-loading on the aircraft increases
## Keep a firm grip on stick while enabled.  Do not be surprised if the temperature of your y-axis motor skyrockets when
## using this effect
gforce_effect_enable = yes                                                                          # DCS
gforce_effect_curvature = 2.2          #adjust the onset characteristics of the effect              # DCS
gforce_effect_max_intensity = 100%                                                                  # DCS
gforce_min_gs = 1.5                    # G's where the effect starts playing                        # DCS
gforce_max_gs = 6.5                    # G limit where the effect maxes out
```

- `gforce_effect_enable` (yes/no)
  - enable or disable the effect
- `gforce_min_gs` (1.0 or greater)
  - the g loading where the effect will start playing
- `gforce_max_gs` (greater than gforce_min_gs)
  - g loading where the strength will reach ‘gforce_effect_max_intensity’
- `gforce_effect_max_intensity` (0.0-1.0 or 0%-100%)

- the maximum force applied in relationship to the constant force slider in VPforce configurator

- `gforce_effect_curvature` (1.0 or greater)
  - affects the onset characteristics of the force effect. A value of 1.0 is a linear increase in force across the defined g range. Increasing the curvature value will result in a flatter increase at the beginning of the range followed by an ever increasing force as the effect approaches the top of the range.
  - Example values (default is 2.2)
  - 

8.7.1.2. Deceleration Effect

Monitors the deceleration g-forces on the aircraft and, if the aircraft is on the ground will apply a forward force (away from pilot) equal to the deceleration G force up to, but not exceeding ‘deceleration_max_force’.

- deceleration_effect_enabled (yes/no, on/off, 1/0)
    - Enable/disable the effect
- deceleration_max_force (0.0-1.0)
    - Controls the maximum force that can be applied

8.7.1.3. Damage Effect (DCS)

Plays a short random direction, random intensity bump each time damage is detected on the aircraft.

**** Can potentially cause performance impact due to large number of calculations required in export script for some aircraft **** Written in a way that will only execute the code in the export script if the feature is enabled in TelemFFB.

- `damage_effect_enabled` (yes/no, 1/0)
    - Controls whether the effect is enabled (both in TelemFFB and the calculations in TelemFFB.lua export script)
- `damage_effect_intensity` (0.0-1.0, %0-%100)
    - Controls the intensity of the generated effects. Note that with the randomized nature of the intensity, some hits will be lower and some higher than the defined value

8.7.1.4. Damage Effect (IL-2)

Plays a short random direction, random intensity bump each time damage is detected on the aircraft.

- `damage_effect_intensity` (0.0-1.0, %0-%100)
    - Controls the intensity of the generated effects. Note that with the randomized nature of the intensity, some hits will be lower and some higher than the defined value

8.7.1.5. AoA Buffeting

8.7.1.6. Runway Rumble/Touchdown

8.7.1.7. Weapons and Countermeasure deployment

8.7.1.8. Speedbrake Motion and Buffeting

8.7.1.9. Gear Motion and Buffeting

8.7.1.10. Flaps Motion

8.7.1.11. Canopy Motion

8.7.1.12. Spoiler Motion and Buffeting

8.7.2. Propeller Aircraft Specific Effects

8.7.2.1. Engine Rumble

8.7.3. Jet Aircraft Specific Effects

8.7.3.1. After Burner Rumble

8.7.3.2. Jet Engine Rumble

8.7.3.3. AoA Reduction Effect

Simulates the increased forward stick pressure that is applied on some fighter aircraft when a critical angle of attack is exceeded. The effect will monitor the AoA and apply a linear force, up to the maximum defined value starting at the ‘start’ AoA and maxing out at the ‘max’ AoA.

- `aoa_reduction_effect_enabled` (yes/no, 1/0)

- `aoa_reduction_max_force` (0.0-1.0, %0-%100)
- `critical_aoa_start` (AoA in degrees)
- `critical_aoa_max` (AoA in degrees)

8.7.4. Helicopter Specific Effects

8.7.4.1. ETL Transition and Overspeed Shake

8.7.4.2. Helicopter Rotor/Engine Rumble

8.7.5. Turboprop Specific Effects (MSFS)

None as of yet

8.7.6. Glider Specific Effects (MSFS)

1.
    2.