

SOFTWARE

ForceSeatPM

SDK

v.1.6

2023.12.06



Contents

1	ForceSeatPM	5
1.1	Introduction	5
1.2	System requirements	6
1.3	Installation	6
1.4	First start	7
1.5	Action Center	7
1.6	Main window and game profiles	9
1.7	Not all profiles have associated games	10
1.8	Games need to be started at least one before they can be configured	10
1.9	Advance users – Platform diagnostic	11
1.10	VR HeadWay	11
1.10.1	Introduction	11
1.10.2	Without VR HeadWay	11
1.10.3	With VR HeadWay	11
1.10.4	VR HeadWay 2.0	12
1.10.5	Configuration - Mapping VR coordinate system	12
1.10.6	Configuration - in-game VR camera zero position	12
1.11	Context help	13
2	SDK	15
2.1	ForceSeatMI	15
2.2	ForceSeatDI	15
2.3	ForceSeatMI vs ForceSeatDI	16
2.4	Features comparison	17
2.5	Documentation	17

ForceSeatPM



1.1 Introduction

ForceSeatPM (platform manager) is professional and specialized software for the platform and is click and play compatible with all major racing and flight titles on PC and constantly updated to support new titles as they are released. The Platform Manager is a powerful engine, yet it's simple to use meaning you can adjust levels of motion, bumps, roll and pitch with a click of a button. The software also includes VR HeadWay which using advanced mathematical formulas to calculate necessary compensation and applies it to the VR headset at run time meaning the most authentic and immersive simulation experience.

The screenshot displays the ForceSeatPM interface, which is divided into several sections:

- Game Selection:** A grid of game thumbnails including 1976, 22 Racing Series, Aerofly FS2, American Truck Simulator, American Truck Simulator ACE, Assetto Corsa, Assetto Corsa ACE, Assetto Corsa Competizione, and Automobilista.
- Configuration Panel (Assetto Corsa):** A detailed settings window for Assetto Corsa, featuring a 3D car model and sliders for various motion effects. The 'Overall' section includes 'Gain' (0.50) and 'Sharpness' (1.00). Other sections include 'Body Roll (Sustained)', 'Body Pitch (Sustained)', and 'Turning Rate', each with its own 'Gain' and 'Sharpness' sliders.
- Platform Diagnostic - ForceSeatPM:** A diagnostic window showing 'Motor platform status is N/A' and a table of motor parameters. The table includes columns for 'Cluster #1: Roll [deg]', 'Cluster #2: Pitch [deg]', 'Motor Position', and 'Motor Speed'. Below the table are sliders for 'Motion (Legacy)', 'Motor Position', 'Sinus Motor Position', and 'Special Effects'.
- Platform Diagnostic - ForceSeatPM (Detailed):** A more detailed diagnostic window showing 'Motor platform status is N/A' and a table of motor parameters. The table includes columns for 'Motor 1 vs 2 Position', 'Motor 1 vs 2 Load [A]', 'Combined Motor Current (1 + 2) [A]', 'Motor 3 vs 4 Position', 'Motor 3 vs 4 Load [A]', 'Combined Motor Current (3 + 4) [A]', 'Motor 5 vs 6 Position', 'Motor 5 vs 6 Load [A]', 'Combined Motor Current (5 + 6) [A]', and 'Aux 1 vs 2 Position'. Below the table are sliders for 'Motion (Legacy)', 'Motor Position', 'Sinus Motor Position', and 'Special Effects'.

1.2 System requirements

Below are system requirements that allows to run ForceSeatPM. Please notice that your configuration also has to meet each game's system requirements.

Supported operating system:

- Microsoft Windows 10 x64
- Microsoft Windows 11 x64

Additionally:

- At least 512MB of free RAM
- At least 220MB free space on system partition
- USB port 2.0 (or faster) or Ethernet port

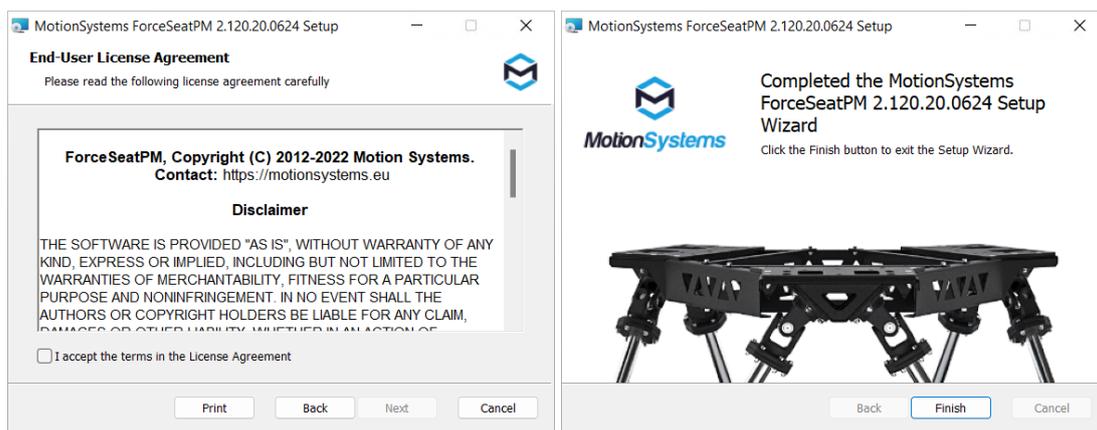
WARNING

Linux-based PCs, Playstation 3/4/5 and X-Box 360/One/Series X are not supported by the platform manager.

1.3 Installation

On Windows 10 or Windows 11, just double click the .msi file.

Then on welcome screen click **Next** button. Read the license, accept it if you agree with terms and click Next button.



During installation, **UAC (User Account Control)** prompt can be displayed. Please accept permission elevation requests.

Finally the program is installed and ready to use. Click Finish, it will start the program.

Please notice that ForceSeatPM is automatically started when you start your computer but it always goes into system tray in that scenario.

1.4 First start

During each start-up ForceSeatPM performs a quick system scan to find installed games. As a result two type of actions are performed:

- When there is a built-in profile for the game, the game executable is automatically assigned to the profile so you can activate given profile and start the game from ForceSeatPM with just one click – **Run game** button.
- When game requires additional plug-in (called also connector), an item is added to **Action Center**. Connectors are not installed automatically as they integrate in the game so you have to begin the process yourself. In most cases admin rights are required to install connector so be prepare for **UAC** prompt.

If after scan there are actions that require your attention, the **Action Center** will be automatically displayed. You can always invoke scan process again manually by opening **Action Center** and clicking **Check configuration**.

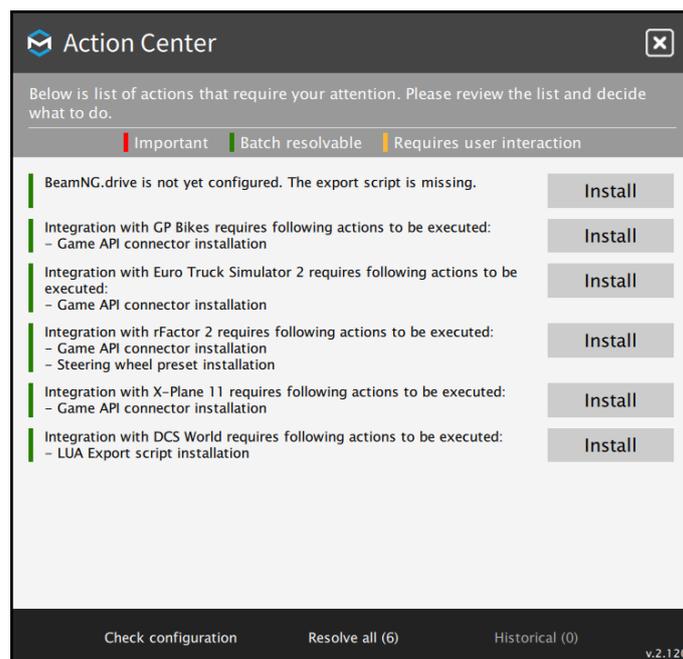
TIP

ForceSeatPM opens a few UDP and TCP ports on localhost in order to get telemetry data from games. This can cause a Windows Firewall (or 3rd party firewall software) pop-up to be displayed. If you block this connections, some games might not work correctly.

1.5 Action Center

It contains a list of issues that require your attention. The list is usually quite long on a first start and depends on number of games you have installed. In most cases it is enough to click the button next to each issue to resolve it.

You can resolve issue by clicking **Install** or you can resolve all issues at once by clicking **Resolve all**.



INFORMATION

You might ask why ForceSeatPM does not install connectors automatically during start-up. This is a very good question and there are two main reasons:

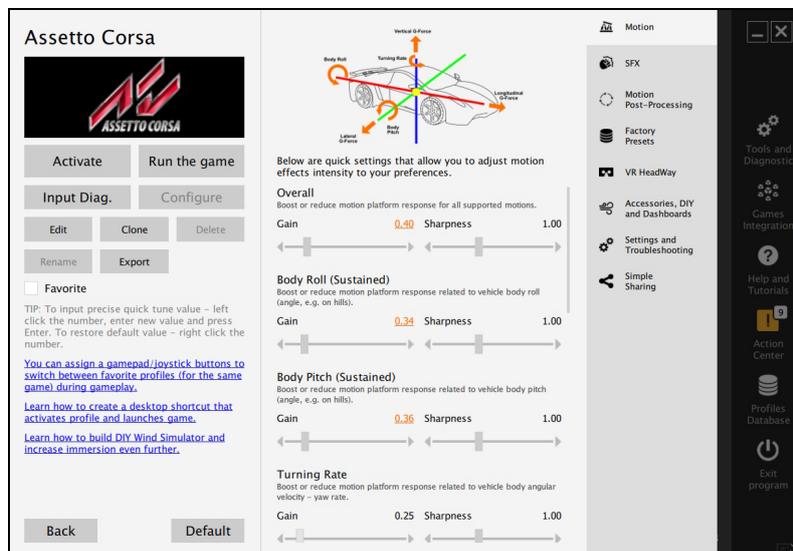
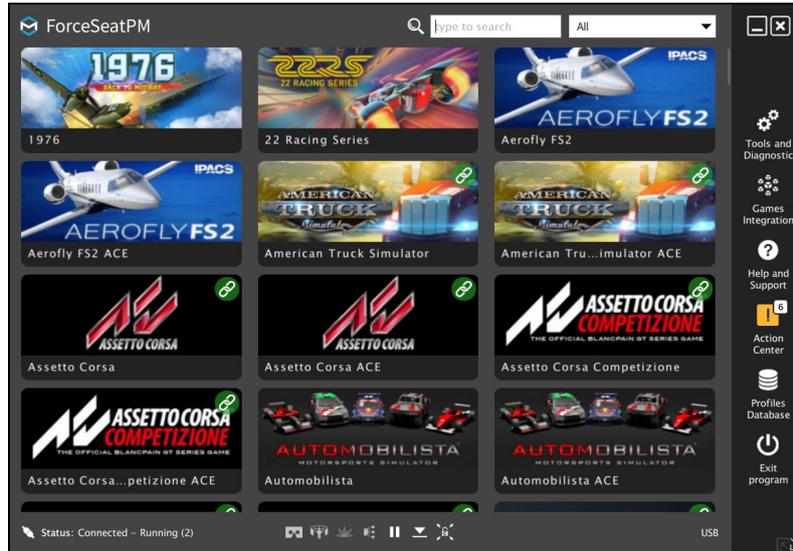
1. Installing connector is a kind of integration within a game (3rd party software). It can break something (e.g. it can cause a conflict with other plug-in). We do not want to perform that kind of operations without informing user about it and without user's confirmation.
2. Usually games are installed in **Program Files**. Copying anything to that folder requires permission elevation. We could display N elevation prompts during startup but this would be confusing for the user – why do they need administrator rights?

Above statements should explain why we have chosen explicit mechanism for connector installation instead of automatic installation. User (you) clicks **Install** and immediately see **UAC** prompt as result of his/her actions.

1.6 Main window and game profiles

Main program window is basically a list of profiles. This is a place where you start your adventure with a new level of entertainment. It also has a quick access toolbar on the right.

Typical scenario is to click on a profile of a game you want to play and then click **Run the game**. It will activate the profile and automatically start the game.



1.7 Not all profiles have associated games

As you have probably noticed, not all profiles have associated games. There are following reasons:

1. Game is not installed in your system.
2. Game is installed but it has not created any registry entries or other files so ForceSeatPM could not find it.
3. Game has been installed after ForceSeatPM has already been started.

In the first case, the only solution is to install the game. In the second case, you can:

1. Click on a profile, then click Edit, go to Game tab and browse for game executable.
2. Click on a profile, then click Activate. Then close (or leave opened) ForceSeatPM main window and start the game from e.g. desktop shortcut.

TIP

Please notice that if you click close icon X, it does not close the ForceSeatPM but only the main window. The program is still available in system tray. In order to completely close the program, you have to click **Exit program** or **Quit** from tray icon menu.

In third case (the game has been installed after ForceSeatPM has already been started) the best idea is to go to **Action Center** and click **Check configuration** – this should find new game and allow to configure it.

1.8 Games need to be started at least one before they can be configured

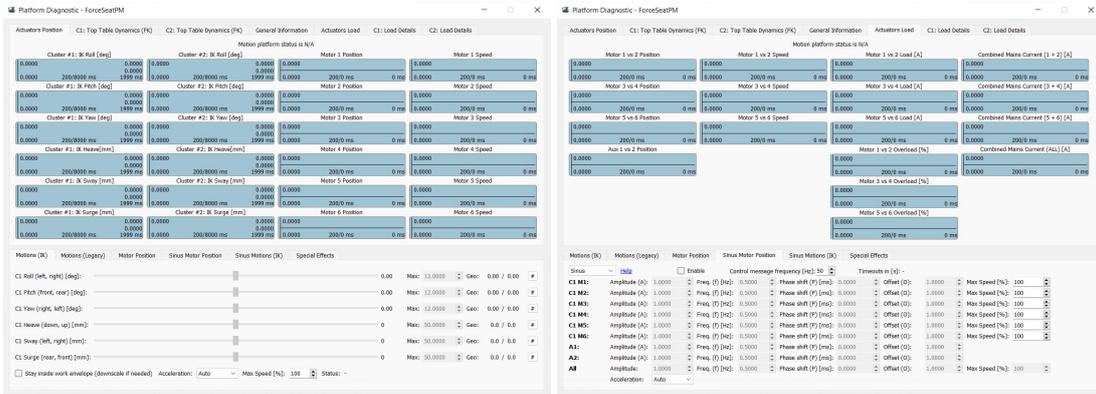
For games like Grid, Grid 2, F1, Dirt, Dirt 2, Dirt 3, the ForceSeatPM is able to find game executable and assign it to profile, but still when you play the game for the first time, there is no platform movements. This problem will be fixed by ForceSeatPM on next system start but it also can be done faster manually.

Basically the game creates a user profile during first start. The ForceSeatPM has to modify this profile in order to enable telemetry data sending via UDP. The correct procedure for these games is as follows:

1. Install the game.
2. Start the game.
3. Create user profile in the game and exit.
4. Go to **ForceSeatPM to Action Center** and click **Check configuration**.

1.9 Advance users – Platform diagnostic

ForceSeatPM has a built-in diagnostic window that allows to control each actuator independently in order to diagnose issues or to check system’s behavior. In order to open diagnostic, go to **Tools and Diagnostic** and then click **Platform Diagnostic**.



1.10 VR HeadWay

1.10.1 Introduction

Using VR headsets with the newest and the most advanced titles may bring you many unforgettable moments. Although everything seems to be natural, photo realistic and almost real, there is one 'but'. Either available VR system on the market is capable to compensate and eliminate movements that come from the motion platform. It may be not obvious for dynamic games where scenery changes every frame but the issue exists.

Movement system generates feeling of the forces that act on your body by providing moves. Those moves are being recognized by VR systems as head movements which is false.

1.10.2 Without VR HeadWay

Taking a turn while driving the car generates force that push the drivers body outside of the seat. To simulate this behavior the motion platform has to lean top frame. These changes the position of the entire cockpit with driver inside. VR system recognizes this as drivers head movements and processes them. Without compensation the drivers head could receive some injuries.

1.10.3 With VR HeadWay

Our Software uses advanced mathematical formulas to calculate necessary compensation and applies it to the VR headset at its run time. It eliminates platform movements from the VR tracking system making your experiences to be more immersive and authentic. Compensation module is transparent for end users, it can be used with almost all our platforms and level of correction can be adjusted to user individual preferences with a few simple sliders.

1.10.4 VR HeadWay 2.0

The next edition of VR HeadWay is a completely new approach to compensation of unwanted movements. Measurements of head position offset relative to the platform pivot point are no longer needed. VR HeadWay 2.0 introduces a new calibration procedure that automatically finds all necessary parameters. As a result, the whole system is easier to configure and it is able to achieve higher compensation precision.

1.10.5 Configuration - Mapping VR coordinate system

This procedure replaces manual head's offset measurements from VR HeadWay 1.0. Settings are stored per VR mode (OpenXR, OpenVR, ...) and per VR origin type chosen by game. It means that sometimes the mapping procedure has to be performed once again if new game is being used. Usually once found settings are still correct even after PC cold restart, but in some cases remapping might be needed after power shutdown. You can redo the VR coordinate system mapping at any time.

TIP

For VR headsets that do not use external reference sensors and are unable to restore the coordinate system after powering off and on, it is required to run the mapping procedure each time the VR system is powered back on.

1. Activate profile for selected game, start the game in VR mode and pause it.
2. Take a seat and put VR headset on your head.
3. Use keyboard or gamepad hotkey to begin VR coordinate system mapping procedure. You can configure binding in **Software Configuration** window.
4. It will take a while, platform will do some moves and the lector will let you know what is going on. Make sure that you are not moving your head during mapping process
5. You can redo the mapping at any time during the gameplay.

1.10.6 Configuration - in-game VR camera zero position

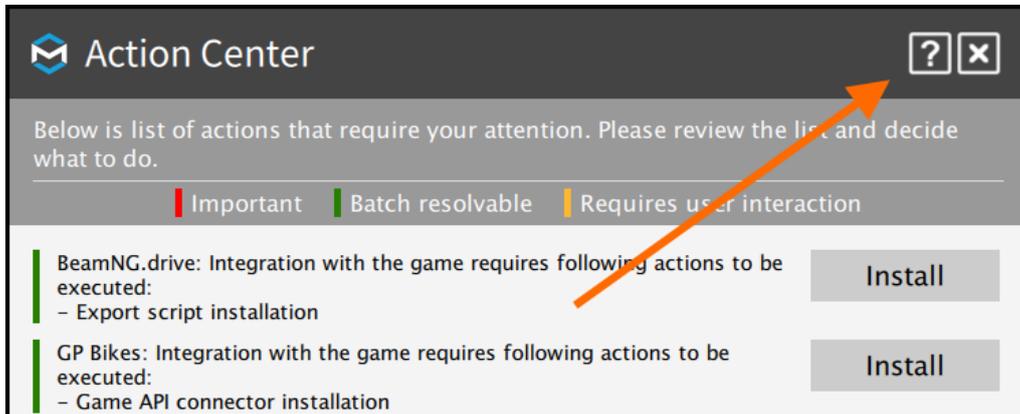
This step should be performed whenever the in-game camera orientation is not correct. Usually, when the game starts, the current position of the VR headset is taken as the zero position. Then the motion platform usually leaves the parked state and the seat moves upwards. As a result, the VR camera in the game is at least shifted vertically. The correction procedure is to lock the motion platform in the center position and reset the in-game camera.

1. Use keyboard or gamepad hotkey to lock the motion platform in centered position. You can configure binding in **Software Configuration** window.
2. Look forward and center VR camera in the game. Check game's user manual for details.
3. Use keyboard or gamepad hotkey to unlock the motion platform from centered position.

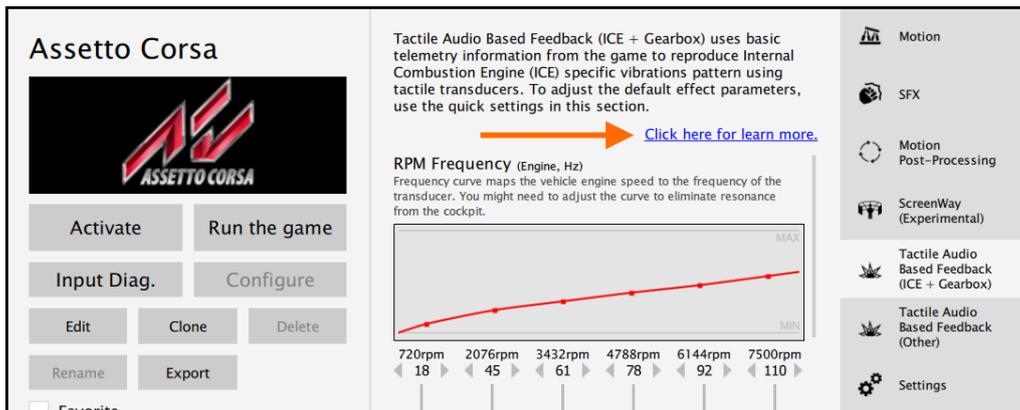
1.11 Context help

Context help is embedded directly in ForceSeatPM and can be accessed on demand. The help covers many topics that are not included in this document and it contains "how to..." tutorials that helps to deal with typical configuration challenges.

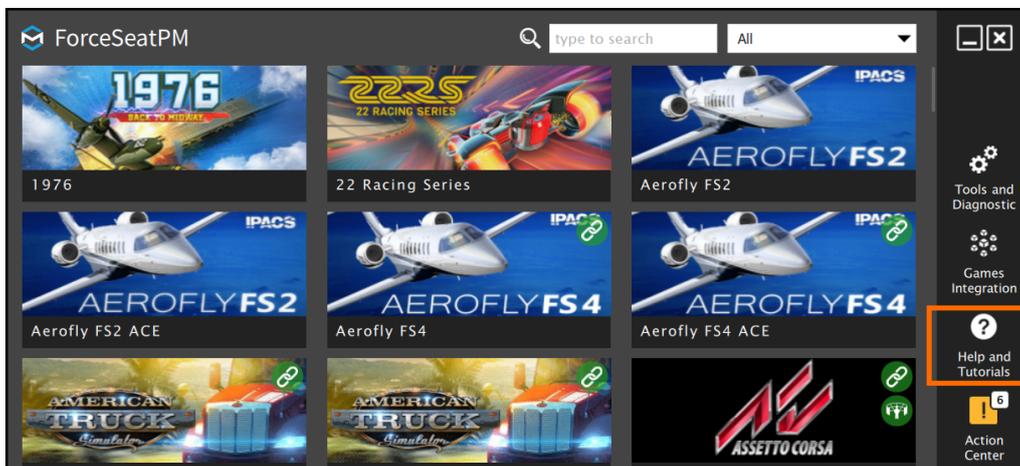
When context help is available for window content, there is **help icon** displayed in the title bar.



If there is a help chapter available for specific topic or section, there is a **link** displayed.



You can also open help browser directly from the main program window.



Help & Tutorials - ForceSeatPM

- Welcome
- Naming Convention
- Motion Post-Processing
- Action Center
- VR HeadWay
- Tactile Audio Based Feedback System
- OBD-II Simulator Configuration
- DIY Wind Simulator with RPM gauge
- How to use Visual Studio Code to edit motion profiles
- How to configure UDP proxy/forwarder for Forza Motorsport 7 and Dash Panel
- How to configure UDP proxy/forwarder for Dirt Rally 2.0 and FanaLEDs
- How to configure UDP proxy/forwarder for F1 2019 and FanaLab
- How to configure UDP proxy/forwarder for F1 22 and SimHub
- How to change profile image
- How to control platform manager from 3rd party software

Motion Post-Processing

Introduction

Telemetry data sent by games is passed to the **Motion Processor**. This module transforms vehicle position, accelerations and velocities into top frame movements of the motion platform. As a result of this operation, 6 values are generated (roll, pitch, yaw, heave, sway and surge). Despite the fact that the motion processor performs necessary filtering on telemetry data, the output data is in the most cases linear compared to the input data (e.g. surge to forward acceleration). Good linearity might be desired for professional motion platforms, but sometimes it is not wanted for less capable motion platforms including seat movers. The **Post Processor** job is to reduce linearity. It uses output from the motion processor, applies required modification (e.g. curvature and gain) and sends output to the motion platform.

```

graph LR
    Game[Game] -- Telemetry --> MP[Motion Processor]
    MP -- Roll --> PP[Post Processor]
    MP -- Pitch --> PP
    MP -- Heave --> PP
    MP -- Sway --> PP
    MP -- Surge --> PP
    MP -- Yaw --> PP
    PP -- Roll --> MP_Platform[Motion Platform]
    PP -- Pitch --> MP_Platform
    PP -- Heave --> MP_Platform
    PP -- Sway --> MP_Platform
    PP -- Surge --> MP_Platform
    PP -- Yaw --> MP_Platform
            
```

Configuration

Both the motion processor and the post-processor can be configured by the user with the usage of quick tunes sliders. In case of the motion processor, list and functions of quick tunes are game dependence. In the case of the post-processor, the list depends on the connected motion platform type and supported type of motions (DoF).

Help & Tutorials - ForceSeatPM

- Motion Post-Processing
- Action Center
- VR HeadWay
- Tactile Audio Based Feedback System
- OBD-II Simulator Configuration
- DIY Wind Simulator with RPM gauge
- How to use Visual Studio Code to edit motion profiles
- How to configure UDP proxy/forwarder for Forza Motorsport 7 and Dash Panel
- How to configure UDP proxy/forwarder for F1 2019 and FanaLEDs
- How to configure UDP proxy/forwarder for F1 22 and SimHub
- How to change profile image
- How to control platform manager from 3rd party software
- SDK - ForceSeatMI
- ACE Editor and Advanced Tuning

3. Press speaker icon to play tune and verify if the mapping is configured properly.
4. You can check which channels are used by effects by in game profile view.

Current System Status

1. The system status is displayed at the bottom of the main window.

SDK

2

2.1 ForceSeatMI

ForceSeatMI is an easy to use yet powerful interface that allows to add a motion platforms support to any application or game (referred as SIM in next sections). In most applications there is no need to control the hardware directly from the SIM. Because of that ForceSeatMI is used only to send telemetry or positioning request to ForceSeatPM. This approach delegates responsibility of transforming telemetry data to an actual platform motion from the SIM to ForceSeatPM. It also simplifies error handling that the SIM has to implement.



INFORMATION

Documentation applies only to ForceSeatMI 2.121 or newer. Older version of the API is not covered by this nor associated documents.

WARNING

ForceSeatMI 2.63 or newer is not backward compatible on interface and binary levels with 2.61 and previous versions. Switching from version 2.61 (or older) to 2.63 (or newer) will require changes in your application source code.

2.2 ForceSeatDI

ForceSeatDI (Direct Interface) is a lower level interface than ForceSeatMI. It controls hardware directly and ForceSeatPM is not required at all. All error handling and status checking have to be performed by the application. This interface allows to control more than one motion platform from the same PC and allows to create complex but fully synchronized movements of multiple motion platforms.



2.3 ForceSeatMI vs ForceSeatDI

ForceSeatMI (Motion Interface) is a programming interface that allows to add a motion platform support to any application or a game. The ForceSeatMI does not control hardware directly – it sends all data to ForceSeatPM. This approach delegates the responsibility of transforming telemetry data to a real motion from the application to ForceSeatPM. It means that application developers do not have to worry about things like platform disconnections, transmission errors, thermal protection warnings or signal filtering.

Direct Interface is less complex than ForceSeatMI. It manages hardware directly, thus ForceSeatPM is not essential. Checking errors or status is a duty of the application. It allows to control multiple platforms from the same computer. Thanks to such a solution it is possible to build complex and fully synchronized movements of multiple motion platforms at once.

WARNING

ForceSeatDI should be used only in very specific applications. For all other applications, ForceSeatMI is recommended.

2.4 Features comparison

	ForceSeatMI	ForceSeatDI
C/C++	■	■
C#	■	■
Python	■	■
Unity 3D	■ (only Windows PC target platform)	■ (only Windows PC target platform)
Unreal Engine	■ (only Windows PC target platform)	■ (only Windows PC target platform)
Matlab/Simulink	■	■
Microsoft Windows	■	■
Linux (Ubuntu 16.04.3 LTS Desktop x64)	□	■
Raspberry Pi 3 (armv7l 4.9.35))	□	■
Raspberry Pi 4 64-bit (armv72 6.1.5))	□	■
Gear VR platforms (e.g. Oculus Go, Samsung Gear VR)	□	□
Multiple platforms from single PC over USB	■ (same data sent to all platforms)	■ (separated control of each platform)
Multiple platforms from single PC over Ethernet	□	■
Easy error handling	■ (by ForceSeatPM)	□ (by the application)
Diagnostic features	■ (by ForceSeatPM)	□ (by the application)
Requires ForceSeatPM	■	□
Telemetry mode & scripting engine	■	□
Motion profile selection by the user	■	□
Inverse kinematics	■	■
Forward kinematics	■	■
Motion compensation for VR	■ (by VR HeadWay in ForceSeatPM)	□ (by the application, e.g. camera position correction)
Licensing	per motion platform (license stored on PC)	per motion platform (license stored on motion controller)

2.5 Documentation

<https://motionsystems.eu/files/ForceSeatMI.pdf>

Detailed description of the ForceSeatMI SDK, including examples.

<https://motionsystems.eu/files/ForceSeatDI.pdf>

Detailed description of the ForceSeatDI SDK, including examples.

https://motionsystems.eu/files/Vehicle_physics_simulation_application.pdf

ForceSeatMI in a vehicle simulation application, including motion cueing description and its tuning capabilities.