

# ORBITER

## Space Flight Simulator



**ORBITER**<sup>2015</sup>  
SPACE FLIGHT SIMULATOR

Orbiter User Manual

# ORBITER User Manual

---

Copyright (c) 2000-2015 Martin Schweiger  
Orbiter home: [orbit.medphys.ucl.ac.uk/](http://orbit.medphys.ucl.ac.uk/)

Wednesday, November 11, 2015

## Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>5</b>
1.1	About Orbiter .....	6
1.2	About this manual .....	6
1.3	Orbiter on the web .....	7
1.4	Finding more help .....	7
1.5	Getting started .....	8
<b>2</b>	<b>WHAT IS NEW IN ORBITER 2015?</b>	<b>9</b>
<b>3</b>	<b>INSTALLATION</b>	<b>11</b>
3.1	Hardware requirements .....	11
3.2	Download .....	11
3.3	Installation .....	11
3.4	Uninstall .....	12
<b>4</b>	<b>BEFORE YOU START: THE LAUNCHPAD</b>	<b>13</b>
4.1	Scenarios tab .....	13
4.2	Parameters tab .....	15
4.3	Visual effects tab .....	16
4.4	Modules tab .....	20
4.5	Video tab .....	21
4.6	Joystick tab .....	22
4.7	Extra tab .....	23
4.8	About Orbiter tab .....	24
<b>5</b>	<b>QUICKSTART</b>	<b>25</b>
<b>6</b>	<b>THE HELP SYSTEM</b>	<b>32</b>
<b>7</b>	<b>KEYBOARD INTERFACE</b>	<b>33</b>
7.1	General .....	33
7.2	Spacecraft controls .....	34
7.3	External camera views .....	35
7.4	Internal (cockpit) view .....	36
7.5	MFD control .....	36
7.6	Menu selections .....	36
<b>8</b>	<b>JOYSTICK INTERFACE</b>	<b>38</b>
<b>9</b>	<b>MOUSE INTERFACE</b>	<b>39</b>
<b>10</b>	<b>SPACECRAFT CLASSES</b>	<b>40</b>
10.1	Delta-glider .....	40
10.2	Shuttle-A .....	40

10.3	Shuttle PB (PTV) .....	42
10.4	Dragonfly .....	43
10.5	Space Shuttle Atlantis .....	43
10.6	International Space Station (ISS) .....	47
10.7	Space Station MIR .....	47
10.8	Lunar Wheel Station .....	48
10.9	Hubble Space Telescope .....	49
10.10	LDEF Satellite .....	50
<b>11</b>	<b>OBJECT INFORMATION .....</b>	<b>51</b>
11.1	Vessel information .....	51
11.2	Spaceport information .....	52
11.3	Celestial body information .....	52
<b>12</b>	<b>CAMERA MODES .....</b>	<b>54</b>
12.1	Internal view .....	54
12.2	External views .....	55
12.3	Selecting the field of view .....	56
12.4	Storing and recalling camera modes .....	57
<b>13</b>	<b>GENERIC COCKPIT VIEW .....</b>	<b>58</b>
13.1	General information display .....	59
13.2	Camera target/mode display .....	60
13.3	Engine information display .....	60
13.4	Navigation mode indicators/controls .....	61
13.5	Surface HUD mode .....	62
13.6	Orbit HUD mode .....	62
13.7	Docking HUD mode .....	62
<b>14</b>	<b>MULTIFUNCTIONAL DISPLAY MODES .....</b>	<b>64</b>
14.1	COM/NAV receiver setup .....	66
14.2	Orbit .....	68
14.3	VOR/VTOL .....	72
14.4	Horizontal Situation Indicator .....	74
14.5	Docking .....	76
14.6	Surface .....	78
14.7	Map .....	81
14.8	Align orbital plane .....	85
14.9	Synchronise orbit .....	86
14.10	RCS Attitude .....	88
14.11	Transfer .....	90
14.12	Ascent profile (custom MFD mode) .....	93
<b>15</b>	<b>SPACECRAFT CONTROLS .....</b>	<b>95</b>
15.1	Main, retro and hover engines .....	95
15.2	Attitude thrusters .....	96
<b>16</b>	<b>RADIO NAVIGATION AIDS .....</b>	<b>98</b>
<b>17</b>	<b>BASIC FLIGHT MANOEUVRES .....</b>	<b>99</b>
17.1	Surface flight .....	99
17.2	Launching into orbit .....	99
17.3	Changing the orbit .....	100
17.4	Rotating the orbital plane .....	101

17.5	Synchronising orbits .....	103
17.6	Landing (runway approach) .....	104
17.7	Docking .....	106
<b>18</b>	<b>FLIGHT RECORDER .....</b>	<b>109</b>
18.1	Playback event editor .....	111
<b>19</b>	<b>SCRIPT INTERFACE .....</b>	<b>113</b>
19.1	Console window .....	113
19.2	Terminal MFD .....	114
19.3	Run a script with a scenario .....	114
19.4	Call a command or script via the API .....	114
<b>20</b>	<b>EXTRA FUNCTIONALITY .....</b>	<b>115</b>
20.1	Scenario editor .....	115
20.2	External MFDs .....	115
20.3	Performance meter .....	116
20.4	Remote vessel control .....	117
20.5	Flight data monitor .....	117
<b>21</b>	<b>FLIGHT CHECKLISTS .....</b>	<b>119</b>
21.1	Mission 1: Delta-glider to ISS .....	119
21.2	Mission 2: ISS to Mir transfer .....	122
21.3	Mission 3: De-orbit from Mir .....	123
<b>22</b>	<b>VISUAL HELPERS .....</b>	<b>125</b>
22.1	Planetarium mode .....	125
22.2	Force vectors .....	127
22.3	Coordinate axes .....	128
<b>23</b>	<b>DEMO MODE .....</b>	<b>129</b>
<b>24</b>	<b>MFD QUICK REFERENCE .....</b>	<b>130</b>
<b>25</b>	<b>SOLAR SYSTEM: CONSTANTS AND PARAMETERS .....</b>	<b>136</b>
25.1	Astrodynamic constants and parameters .....	136
25.2	Planetary mean orbits (J2000) .....	136
25.3	Planetary orbital element centennial rates .....	137
25.4	Planets: Selected physical parameters .....	137
25.5	Rotation elements .....	138
25.6	Atmospheric parameters .....	138
<b>26</b>	<b>CALCULATION OF ORBITAL ELEMENTS .....</b>	<b>140</b>
26.1	Calculating elements from state vectors .....	140
<b>27</b>	<b>TERMS OF USE .....</b>	<b>143</b>
27.1	Orbiter Freeware License .....	143
27.2	Disclaimer of warranty .....	143

## 1 Introduction



### Welcome to Orbiter 2015!

Orbiter 2015 is the latest instalment of the Orbiter Spaceflight simulator series. The most notable improvement from the 2010 Edition is the added support for surface elevation of planetary bodies, higher resolution surface and cloud textures, and improved surface collision modelling.

High-resolution planetary texture packs are available for Earth, the Moon and Mars. In fact, there is now so much texture and elevation data available that you may need to free a bit of space on your hard disk to install it all – but I hope you'll agree that the result is worth the extra space.

*Martin Schweiger*

## 1.1 About Orbiter

They say a little knowledge is a dangerous thing, but it's not one half so bad as a lot of ignorance.

*Terry Pratchett – Equal Rites*

Orbiter is a space flight simulator based on Newtonian mechanics. Its playground is our solar system with many of its major bodies - the sun, planets and moons. You take control of a spacecraft - either historic, hypothetical, or purely science fiction. Orbiter is unlike most commercial computer games with a space theme - there are no predefined missions to complete (except the ones you set yourself), no aliens to destroy and no goods to trade. Instead, you will get a pretty good idea about what is involved in real space flight - how to plan an ascent into orbit, how to rendezvous with a space station, or how to fly to another planet. It is more difficult, but also more of a challenge. Some people get hooked, others get bored. Finding out for yourself is easy - simply give it a try. Orbiter is free, so you don't need to invest more than a bit of your spare time.

Orbiter is a community project. The Orbiter core is just the skeleton that defines the rules of the simulated world (the *physical model*). A basic solar system and some spacecraft (real and fictional) are included, but you can get a lot more with add-on modules developed by other enthusiasts in the Orbiter community. There are add-ons for nearly every spacecraft that ever flew (and quite a few that never got beyond the drawing board), for many more celestial bodies in the solar system (or entirely new fictional systems), for enhanced instruments, and much more. The Orbiter web site contains links to many Orbiter add-on repositories.

## 1.2 About this manual

This document is the main help file that comes with the basic distribution of Orbiter. It is a User's Guide to the Orbiter software - which is to say that it gives an introduction into *how* most things work, but doesn't tell you much *why* they behave as they do. By following the instructions, you will find out how to operate the engines of your spacecraft, how to use the instruments, and how to perform the most common missions.

But a big part of the appeal of Orbiter is finding out about the *why* - why do spacecraft in orbit behave as they do, what is involved in a gravity-assist flyby, why do rockets have multiple stages, why can it be tricky to line up for docking with a space station, what do the numbers in the instrument displays actually mean ... ?

This is where physics comes into the picture. If you want to conquer the final frontier, you will at some stage need to understand a few of the fundamental physical concepts that form the basis of astrodynamics and space flight. Luckily most of it is not very difficult - if you learn a bit about forces and gravity ("Newtonian mechanics") and how they relate to the motion of planets and spacecraft in orbit ("Kepler's laws"), you will have covered a good deal of it. Of course, there are always opportunities to dig

deeper into the details, so your next steps might be finding out about the effects of orbit perturbations, attitude control, trajectory optimisation, mission planning, instrument design - to name just a few.

Don't get frustrated if you don't succeed immediately – it's only rocket science. Read the documentation and try some of the numerous Orbiter tutorials available on the internet, and you will soon be orbiting like a pro.

Eventually you might start to develop your own add-on modules to enhance Orbiter's functionality, write tutorials and help files for newcomers - or even take active part in the Orbiter core development by identifying and discussing flaws or omissions in the Orbiter physics model (and there are still many!)

### 1.3 Orbiter on the web

The Orbiter home page can be found at [orbit.medphys.ucl.ac.uk/](http://orbit.medphys.ucl.ac.uk/). It is your portal to Orbiter news, downloads, forum, addon sites, and related pages.

The main Orbiter forum, [www.orbiter-forum.com/](http://www.orbiter-forum.com/), is a friendly meeting place for an active community of new and seasoned users and developers. It is a good place to find answers to any problems you may encounter, or just to hang out with fellow Orbonauts. Suggestions, bug reports (and of course praise) are always welcome. Links to other forum sites can be found on the Orbiter web site.

Next door to the forum, at [www.orbithangar.com](http://www.orbithangar.com), is the primary Orbiter add-on repository, where you can find a huge number of user-created spacecraft, instruments, textures, and more. And once you have started to write your own plug-ins, you can upload them here to share with others.

The Orbiter wiki, at [www.orbiterwiki.org/wiki/Main\\_Page](http://www.orbiterwiki.org/wiki/Main_Page), is a community-maintained site which contains useful information for users and developers.

For general information about Orbiter, have a look at the Wikipedia entry, [en.wikipedia.org/wiki/Orbiter\\_\(sim\)](http://en.wikipedia.org/wiki/Orbiter_(sim)).

A site dedicated to Orbiter graphics development is the Orbiter Visualisation Project at [sourceforge.net/projects/orbitervis/](http://sourceforge.net/projects/orbitervis/).

### 1.4 Finding more help

The help files that come with the main Orbiter package are located in the *Doc* sub-folder below your main Orbiter directory. Many add-ons will place their own help files in the same directory after installation. The *Doc\Technotes* folder contains some documents with technical details and background information for interested readers. They are not required for using Orbiter.

Many people have written documentation and tutorials covering particular aspects of Orbiter. Links can be found on the *Related sites* page of the Orbiter home page.

A very good introduction to using and understanding Orbiter for beginners (and a handy refresher for old-timers) is Bruce Irving's online book *Go Play In Space*, which can be found via a link from the *Manual* page on the Orbiter web site.

The scientific and technical background of space flight is covered in many textbooks and online sites. A good introduction is JPL's [Basics of Space Flight](#), or R. Braeunig's [Rocket & Space Technology](#). Among the many online resources for the general mathematics and physics relevant for space flight, you might find the Scienceworld site useful, at [scienceworld.wolfram.com](#).

## 1.5 Getting started

If you are a first-time user, it is probably a good idea to have a look at this manual to get you off the ground quickly. Ideally, use it together with the simulator. If you don't want to print it, run Orbiter in window mode (see Section 4.5) and have the manual open next to it.

For installation help, see Section 3. The first time you run Orbiter, you will have to configure the video options (Sec. 4.5). Then you are good to go – see Sec. 4.1 on how to select a scenario and launch the simulation.

To get a feel for Orbiter, you can run some of the pre-recorded flights and tutorials. These are the scenarios you find under the *Tutorials* and *Playback* folders. They don't require any user input, so you can lean back and enjoy the view.

Once you are ready to take control, have a look at the *Quickstart* chapter (Sec. 5). It contains step-by-step instructions for takeoff, flight and landing in the futuristic Delta-glider.

Some more complex missions, including a flight from the Kennedy Space Center to the International Space Station, can be found in *Flight checklists* folder (Sec. 21).

For an overview of basic spacecraft controls, see Sec. 15. A detailed list of common keyboard commands can be found in Sec. 7.

And once you have made your first steps into orbit, you might want to consult the rest of the manual to learn about some of the more advanced details of Orbiter.

## 2 What is new in Orbiter 2015?

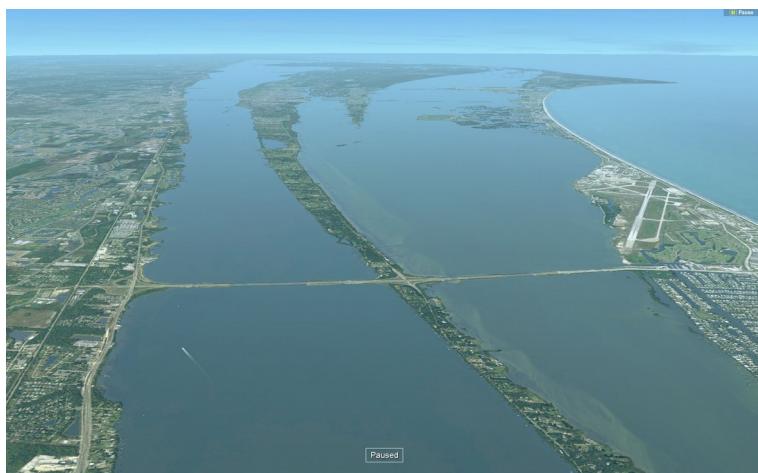
### Terrain elevation support

Orbiter now supports terrain elevation for planetary bodies. Planets are no longer perfect spheres – mountain ranges and canyons are now modelled properly. The default Orbiter installation comes with elevation data for Earth, the Moon, Mars and some minor bodies. Expect other bodies to be supported in addons soon.



*The Alps in Orbiter  
2015: Terrain support  
with SRTM-90 DEM  
data.*

In addition to the new elevation support, Orbiter now also provides a new texture format that can display planetary surfaces and cloud layers at much higher resolution than before (~1 pixel/m for Earth). Global Earth textures have been generated from raw Landsat-7 scenes with extensive processing for colour correction, seam and cloud removal and coastline fitting. For Florida and the Kennedy Space Center, even higher resolution textures have been generated from aerial image sources. Moon, Mars and several other planetary bodies also support the new texture format in the default installation.



*High-resolution textures:  
View towards Kennedy  
Space Center from the  
south. Patrick AFB is  
visible on the right.*

The texture packages are now too large to be included with the basic Orbiter installation. The default Orbiter package contains low-resolution versions for all bodies. You

can download and install the high-resolution versions from the Orbiter texture repository.

### Improved terrain impact model

Vessel impact on planetary surfaces is now modelled more realistically, using a dynamic compression model. Vessels can now define a convex hull of impact points to provide surface interaction parameters at all attitudes. Impact hulls for docked vessels are now automatically merged, allowing docked assemblies on the ground.

### New cockpit models for default vessels

The Delta-glider and Shuttle-A vessels included in the Orbiter basic installation now come with improved virtual cockpits and 2D instrument panels. The Delta-glider has a few interesting new features like the automatic engine gimbal mode, hover attitude control, and a pressure control module in the overhead panel.



*Delta-glider virtual cockpit view.*

Orbiter 2015 contains a re-worked physics engine with improved accuracy. Orbiter now automatically subsamples frame updates when required. Subsampling is particularly important during impact events, where large forces are active over very short time intervals.

### And much more ...

A lot of additional features and bug fixes have gone into the new version to improve overall stability and user experience. And of course, development is going on, with additional fixes to be distributed in patch releases whenever necessary.

## 3 Installation

This section lists the computer hardware requirements for running Orbiter, and contains download and installation instructions.

### 3.1 Hardware requirements

Orbiter should run on most Windows PCs with 4GB of memory or more, running Windows XP or later. A high-performance graphics card is recommended for improved performance.

The basic installation requires approximately 2GB of hard disk space. Installing high-resolution texture packs can significantly increase space requirements. A complete installation of all currently available texture packs may require close to 100GB of space.

Orbiter supports an optional joystick.

The TrackIR head tracker is supported via a plugin module included in the basic installation.

### 3.2 Download

The Orbiter distribution can be obtained from one of several Orbiter mirror sites on the internet. You can find links to these mirrors at the Download page of the Orbiter site, <http://orbit.medphys.ucl.ac.uk/>. Orbiter is distributed in several compressed software packages (.zip files). The *Base* package contains the basic Orbiter system and is the only required package. All other packages are optional extensions to the basic system.

All package names contain a 6-digit time stamp (YYMMDD) identifying the modification date of the package. For example, *orbiter060504\_base.zip* contains the base package built on May 4, 2006. Note that not all current packages may have the same time stamp. In particular, high-resolution planetary texture packages are rarely updated and may have an older time stamp. Check the download pages for the latest versions of all packages.

### 3.3 Installation

- ▶ The Orbiter installation package comes in two flavours: an MSI (Microsoft installation) package, and a ZIP package.
- ▶ The MSI package allows to install Orbiter like a standard Windows software package. It guides the user through the installation process with a wizard dialog, and provides a desktop icon and uninstall option. This method is recommended for general users.
- ▶ The ZIP package requires manual extraction of the contents into a new folder. This method allows installation of multiple Orbiter installations, and is non-intrusive (it doesn't modify the registry and doesn't install anything outside the Or-

biter directory. However, the user is responsible for providing the runtime environment, including required system libraries. This method is recommended for advanced users. For the ZIP installation,

- Create a new folder for the Orbiter installation, e.g. c:\Orbiter20105 or %HOMEPATH%\Orbiter2015. Note that creating an Orbiter folder in Program files or Program files (x86) is not recommended, because Windows puts some restrictions on these locations.
  - If a previous version of Orbiter is already installed on your computer, you should not install the new version into the same folder, because this could lead to file conflicts. You may want to keep your old installation until you have made sure that the latest version works without problems. Multiple Orbiter installations can exist on the same computer.
  - Unzip the Orbiter ZIP installation package into the new folder, using either the default Windows unzip function, or an external tool like 7-zip or WinZip. Important: Take care to preserve the directory structure of the package (for example, in WinZip this requires to activate the “Use Folder Names” option).
  - After unzipping the package, make sure your Orbiter folder contains the executable (orbiter.exe) and, among other files, the Config, Meshes, Scenarios and Textures subfolders.
- Run orbiter.exe. This will bring up the Orbiter “Launchpad” dialog, where you can select video options and simulation parameters.
- You are now ready to start Orbiter. Select a scenario from the Launchpad dialog, and click the “Launch Orbiter” button!



If Orbiter does not show any scenarios in the *Scenario* tab, or if planets appear plain white without any textures when running the simulation, the most likely reason is that the packages were not properly unpacked. Make sure your Orbiter folder contains the subfolders as described above. If necessary, you may have to repeat the installation process.

### 3.4 Uninstall

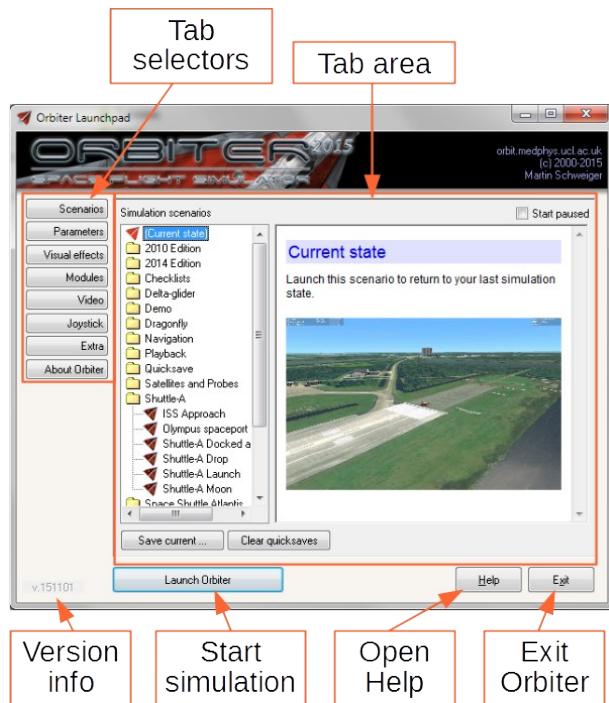
If you installed the MSI package, simply use the standard Windows method for uninstalling a software package. Note that this will only remove the files of the original installation package. Texture packs and addons installed later may need to be deleted manually from the Orbiter folder.

If you installed the ZIP package, simply remove the Orbiter folder with all contents and subdirectories. This will completely remove Orbiter from your hard drive.

## 4 Before you start: The Launchpad

Starting Orbiter.exe brings up the *Orbiter Launchpad* dialog box. The launchpad is your gateway to Orbiter. From here, you can

- ▶ select and launch a simulation scenario
- ▶ set simulation, video and joystick parameters
- ▶ load available plug-in modules to extend the basic Orbiter functionality
- ▶ open the online help system
- ▶ launch the Orbiter simulation window, or
- ▶ exit to the desktop



*Orbiter Launchpad dialog: Start a simulation scenario, edit parameters and video options, activate modules.*

Clicking on one of the tab selector buttons along the left edge of the dialog box opens the corresponding configuration page.

Important: Before running Orbiter for the first time, make sure that all simulation parameters (in particular the video options) are set correctly.

When you are ready, select a scenario, and press the "Launch Orbiter" button to jump into the simulation.

### 4.1 Scenarios tab

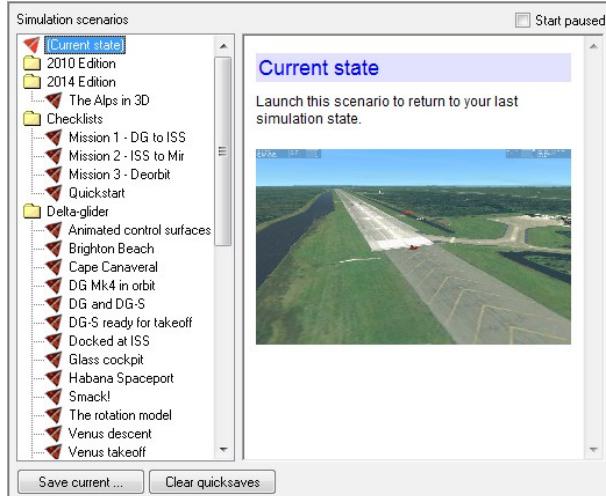
The *Scenarios tab* allows you to manage and browse the available simulation startup scenarios. A "scenario" defines the initial setup of a simulation session (the date, spacecraft positions, velocities and other parameters).

The scenario list contains all stored scenarios (including any you created yourself) in a hierarchical folder structure. Double-click on a folder to open its contents. Double-click on a scenario (marked by the red "Delta-glider" icon) to launch it.

Selecting a scenario or folder brings up a description on the right of the dialog box. This can include mission objectives, screenshots, and background information.

There are a few special scenarios and folders:

- ▶ The (**Current state**) scenario is automatically generated whenever you exit the simulator. Use this to continue from the latest exit state.
- ▶ The **Tutorials** folder contains pre-recorded flights with onscreen annotations that explain different aspects and stages of space flight missions.
- ▶ The **Playback** folder contains the flights you have recorded with Orbiter's built-in flight recorder. Launching one of these will start a replay.
- ▶ The **Quicksave** folder contains in-game saved scenarios generated by pressing **Ctrl S**. Multiple quicksaves are possible. Orbiter saves the quicksave states under the original scenario name, followed by a quicksave counter. The counter is reset each time the simulation is launched, so make sure to copy any scenarios you want to keep!
- ▶ The **Demo** folder can be filled with scenarios that are automatically run in kiosk/demo mode (see Section 22.2). This allows to put together a set of simulations that can be run in unsupervised environments.



### To start the simulation paused:

Tick the *Start paused* box to pause the simulation on start. You can resume the simulation by pressing **Ctrl P**.

### To save your own scenarios:

After exiting a simulation session, click the *Save current* button to save the current simulation state in a new scenario file. For setting up custom simulation scenarios, see also the *Scenario Editor Manual* (ScenarioEditor.pdf).

### To clear quicksaved scenarios:

Click the *Clear quicksaves* button to delete all scenarios stored in the **Quicksave** folder.

## 4.2 Parameters tab

The *Parameters* tab contains various options to customise the simulation behaviour, including realism and difficulty settings, background star rendering, instrument display settings, and focus mode for dialog boxes.

### Realism

- ▶ **Complex flight model:** Select the realism of the flight model for spacecraft. Tick this box to enable the most realistic flight parameters available for all vessel types. Disabling this option may activate simplified flight parameters which make spacecraft easier control for newcomers. Not all vessel types may support this option.
- ▶ **Damage and failure simulation:** Spacecraft can sustain damage and system failure, for example if operational limits are exceeded. Not all vessel types may support this option.
- ▶ **Limited fuel:** Un-tick this box to ignore fuel consumption of your spacecraft.

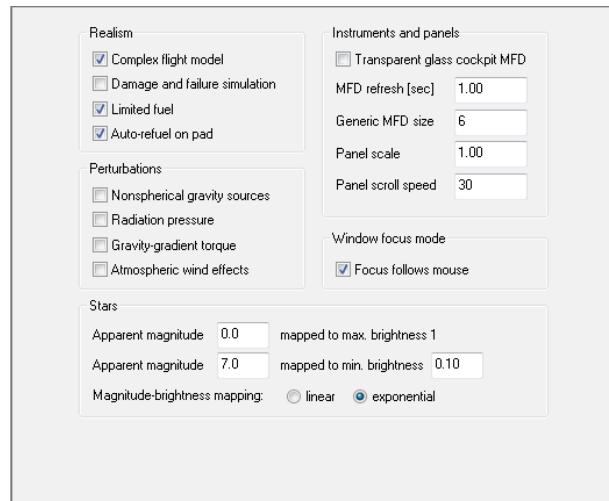


Some of the more “realistic” spacecraft, such as the Space Shuttle, may *NOT* work properly if “Limited fuel” is not selected, because they rely on the reduction of mass during liftoff as a consequence of fuel consumption.

- ▶ **Nonspherical gravity sources:** This option activates a more complex gravity calculation which can take into account perturbations in the gravitational potential due to nonspherical object shapes, thus allowing more accurate orbit predictions. Note that this option can make orbital calculations more difficult, and may reduce the stability of instruments that don't take this effect into account. For a planet to make use of the perturbation code, its configuration file must contain the `JCoeff` entry. For background and technical implementation details please refer to the Orbiter Technical Note [Doc/Technotes/Gravity](#).
- ▶ **Gravity-gradient torque:** If this option is enabled, vessels can experience an angular moment in the presence of a gravitational field gradient. This will be noticeable in particular in low orbits and can lead to attitude oscillations around the equilibrium or attitude-locked orbits. For background and technical implementation details please refer to the Orbiter Technical note [Doc/Technotes/Distmass](#).

### Window focus mode

- ▶ **Focus follows mouse:** If this option is ticked, the input focus is switched between the Orbiter simulation window and any open dialog boxes by moving the mouse over the window. If unticked, the focus is switched in normal Windows style by clicking the window.



## Stars

The parameters in this group the number and brightness of background stars displayed on the celestial sphere. Orbiter uses the Hipparcos star catalogue with more than  $10^5$  entries.

The "apparent magnitude" is a logarithmic scale describing the brightness of a star as seen from Earth. The brightest star (except for the sun), Sirius, has an apparent magnitude of  $m_v = -1.5$ . The faintest stars visible without instruments are approximately of magnitude  $m_v = 6$ .

Using a higher magnitude value for the max. brightness setting will render stars brighter. Using a higher magnitude for the min. brightness setting will increase the number of faint stars rendered. Increasing the min. brightness level will make faint stars look brighter.

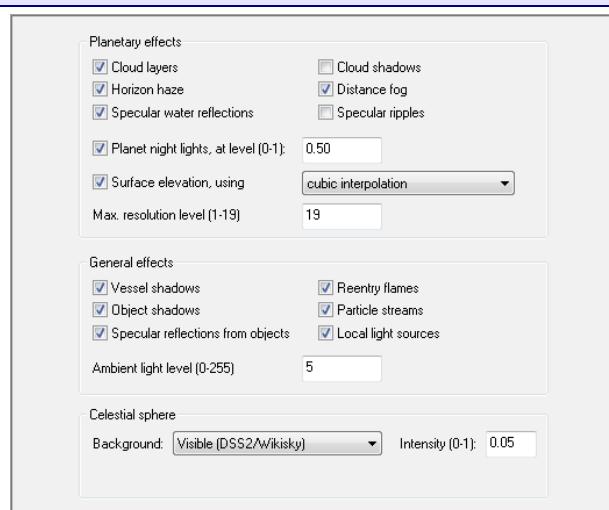
Using logarithmic mapping will increase the contrast between bright and faint stars to a more realistic level.

## Instruments

- ▶ **Transparent MFD:** Make the onscreen multifunctional displays transparent. This provides a better view of the 3D environment, but makes it more difficult to read the instruments.
- ▶ **MFD refresh:** Time (in seconds) between MFD updates. Shorter intervals provide smoother updates, but may degrade performance. Some built-in MFD modes, such as the Surface and HSI modes, define a lower limit for the update frequency.
- ▶ **Panel scale:** Sizing factor for instrument panels. Scale 1 provides optimal visual quality, but other values may be used to adapt the panel size at low or high screen resolutions.
- ▶ **Panel scroll speed:** Determines how fast the panel can be scrolled across the screen [pixels/second]. Negative values invert the panel scroll direction.

## 4.3 Visual effects tab

The *Visual effects* tab provides options for tuning the rendering parameters and graphic detail. These options will improve the visual appearance and realism of the simulator, but most of them can have an adverse effect on simulation performance (frame rates) when enabled, and may increase video and main memory demands, so they should be used with care, in particular on less powerful computers. As a first step



in troubleshooting Orbiter problems, it is often a good idea to turn off all visual effects.

Note that some advanced rendering options can also be found in the *Extra* tab, under *Visualisation parameters*. This includes mipmap and anisotropic filtering options as well as the new on-demand texture loading feature.

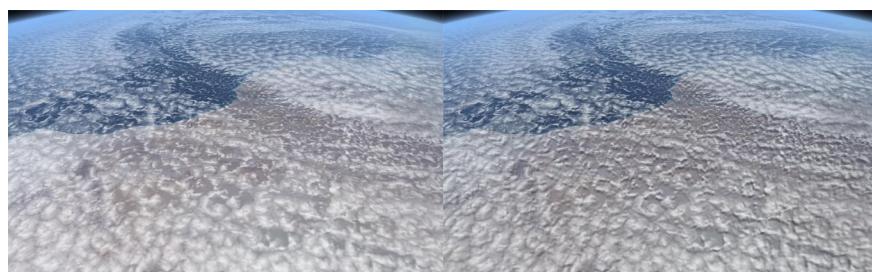
### Planetary effects

- ▶ **Cloud layers:** Render clouds as a separate mesh layer for appropriate planets.



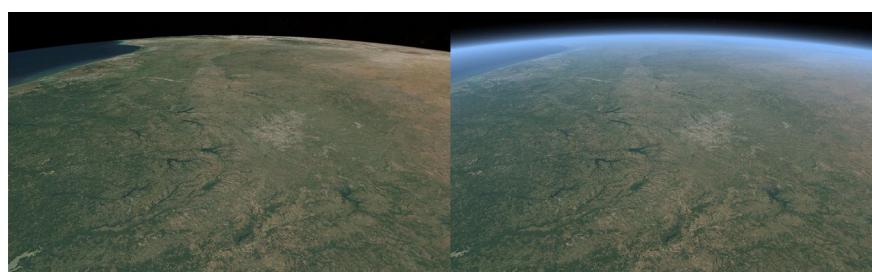
*Cloud layers disabled (left) and enabled (right).*

- ▶ **Cloud shadows:** Render cloud shadows cast on the planet surface. Only planets whose config files contain a *CloudShadowDepth* entry  $< 1$  will actually render cloud shadows.



*Cloud shadows disabled (left) and enabled (right).*

- ▶ **Horizon haze:** Render intensity-graded (“glowing”) horizon layer for planets with atmospheres.

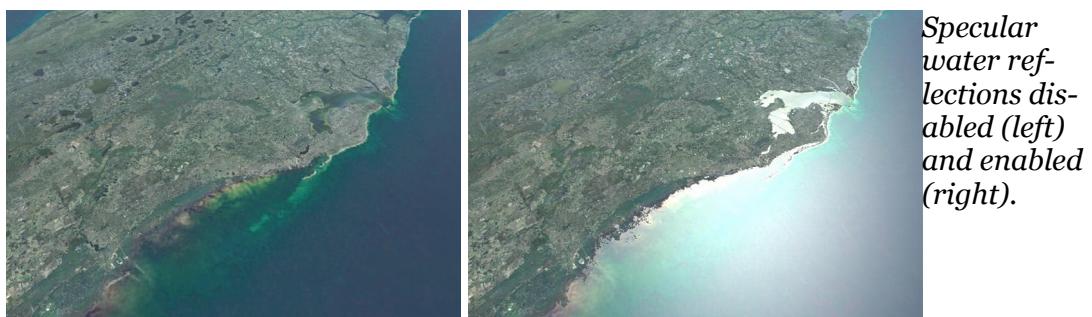


*Horizon haze disabled (left) and enabled (right).*

- ▶ **Distance fog:** Apply atmospheric mist and fog effects to distant object when viewed through planetary atmospheres.



- ▶ **Specular water reflections:** Render water surfaces on planets with specular reflection effects.



- ▶ **Specular ripples:** Generate “ripple” effect in specular reflections from oceans for improved appearance of water surfaces.



- ▶ **Planet night lights:** Render city lights on the dark side of planet surfaces where available.



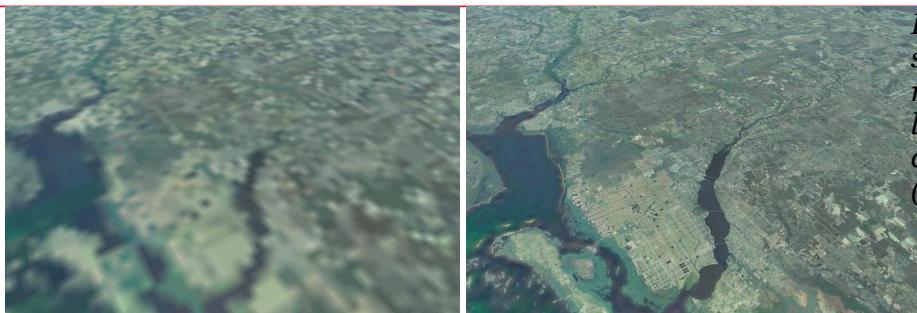
*Planet night lights disabled (left) and enabled (right).*

- ▶ **Night light level:** Defines the brightness of night city lights. Valid range is 0 to 1. (ignored if planet night lights are disabled)
- ▶ **Max. resolution level:** The maximum resolution at which planetary surfaces can be rendered. Supported values are 1 to 14. Higher values provide better visual appearance of planets that support high texture resolutions, but also significantly increase the demand on computing resources (graphics processor and memory). Note that the actual resolution level supported by any planetary body may be lower than this value, depending on the texture set available. Higher resolution textures for many bodies may be downloaded from the Orbiter website or add-on repositories. The highest resolution levels are usually only supported in selected areas of the surface (e.g. around spaceports).



If you are using many high-resolution texture maps, it is important to activate the load-on-demand feature, to avoid excessive loading and closing times. This feature can be activated under the *Extra* tab of the Orbiter Launchpad: Select

*Visualisation parameters → Planet rendering options → Load on demand*



*Florida scenery at resolution level 10 (left) and 14 (right).*

## General effects

- ▶ **Vessel shadows:** Enable shadows cast by spacecraft on planet surfaces.
- ▶ **Object shadows:** Enable dynamic shadows of ground-based objects such as buildings.
- ▶ **Specular reflections from objects:** Render reflective surfaces like solar panels, window panes or metallic surfaces. May degrade performance.
- ▶ **Reentry flames:** Render glowing plasma hull during reentry.

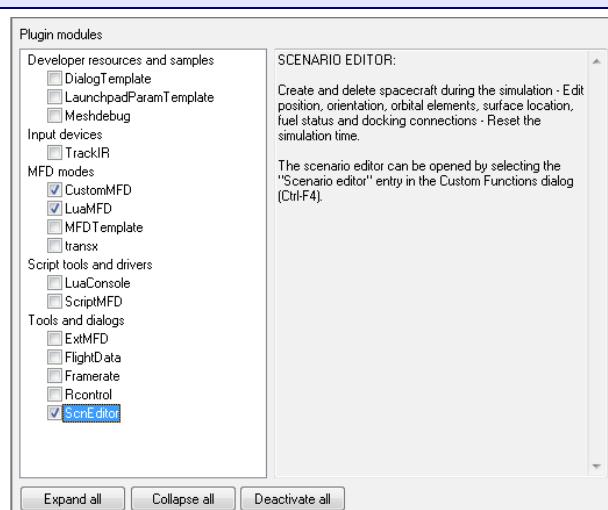
- ▶ **Particle streams:** Render ionised exhaust gases and vapour trails with particle effects.
- ▶ **Local light sources:** Enable localised light sources, e.g. from engines, landing lights, floodlights, etc. This option can have a significant influence on frame rates.
- ▶ **Ambient light level:** Defines the brightness of the unlit side of planets and moons. Ambient level 0 is the most realistic, but makes it difficult to spot objects in the dark. Level 255 is uniform lighting (no darkness).

### Celestial sphere

- ▶ **Background:** Select a bitmap to cover the celestial sphere background. Various options are available in the default distribution, including sky surveys from various mapping projects at different wavelength ranges. More maps may be available as addons.
- ▶ **Intensity:** The brightness of the background image (range: 0-1). For a realistic setting, try the *Visible* map with a very low intensity setting (e.g. 0.05).

## 4.4 Modules tab

The *Modules* tab allows the activation and deactivation of plug-in modules for Orbiter which can extend the functionality of the core simulator. Plug-ins can contain additional instruments, dialogs, interfaces to external programs, etc. Make sure you only activate modules you actually want to use, because modules can take up some processing time even if they run in the background, and thus affect Orbiter's performance.



To activate a module, click the tick box next to its entry in the list. By clicking on the entry itself, many modules provide a short description about their function and user interface in the right panel. Entries are grouped in categories. You can expand or collapse categories by double-clicking the category header. The buttons at the bottom of the tab allow expanding or collapsing the entire list, and quick deactivation of all modules.

The modules provided with the standard Orbiter distribution are demos from the SDK package, and are available in full source code. A wide variety of additional modules by 3<sup>rd</sup> party add-on developers can be downloaded from Orbiter repositories on the internet.

Some of the standard modules distributed with Orbiter are:

**ScnEditor:** A versatile scenario editor that allows adding, editing and deleting spacecraft in a running simulation. See Section 20.1 for more details.

**ExtMFD:** This module allows to open additional multifunctional displays in external dialog boxes. Useful if you need more information than a vessel's built-in MFD displays provide, or if you want to track flight data in external camera views.

**CustomMFD:** This module provides an additional “Ascent MFD” mode for the multifunctional displays, which can be selected via **[Shift F1] - [Shift P]**.

**Rcontrol:** Remote control of ship engines. This allows to manipulate vessels even if they don't have input focus. If this module is active, the remote control window can be selected from the *Custom Functions* list (**[Ctrl F4]**).

**FlightData:** Real-time atmospheric flight data telemetry. If this module is active, the flight data window can be selected from the *Custom Functions* list (**[Ctrl F4]**).

**Framerate:** A graphical simulation frame rate (FPS) display. If this module is active, the frame rate window can be selected from the *Custom Functions* list (**[Ctrl F4]**).

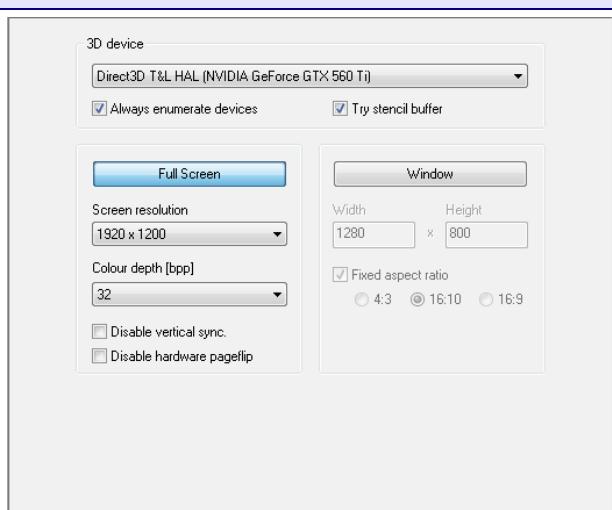
**LuaConsole:** Provides a console window for interactive processing of script commands from the *Custom Functions* list.

**LuaMFD:** Adds a new MFD mode for script input via a console MFD.

## 4.5 Video tab

The *Video tab* provides options to select the rendering device, switch between full-screen and windowed mode, and set the resolution, window size and colour depth.

**3D Device:** Lists the available hardware and software devices for 3D rendering. Select a hardware device with transform and lighting capabilities when possible, such as *Direct3D T&L HAL* or similar. (On some systems, the hardware devices might be listed with the name of your graphics card). Software devices such as *RGB Emulation* will produce poor performance. Note that some hardware devices do not support window mode.



**Always enumerate devices:** Tick this box if Orbiter does not display 3D devices or screen modes correctly. This option enforces a hardware scan whenever Orbiter is launched and skips the device data stored in device.dat. **Make sure to tick this box after upgrading your graphics hardware or DirectX/video drivers to make Orbiter aware of the changes.**

**Try stencil buffer:** Enables stencil buffering, if the video mode supports it. Stencil buffers can improve various visual effects (for example, provide support for alpha-

blended shadows), but may have a slight impact on frame rates. If the selected video mode doesn't support stencil buffers, this option is ignored.

**Full Screen:** Select this option to run Orbiter in full-screen mode. You can choose the screen resolution and colour depth from the lists provided. Only modes supported by the selected device are listed here. Higher resolution and colour depth will improve the visual appearance at the cost of reduced performance.

In addition, you can select the *Disable vertical sync* option. This allows Orbiter to update a frame without waiting for a synchronisation signal from the monitor. This can improve frame rates, but may lead to visual artefacts (tearing).

On some systems the hardware frame buffer switching may cause the screen occasionally to flash white. Use *Disable hardware pageflip* to solve this problem. Disabling hardware pageflip also disables vertical sync.

**Window:** Select this option to run Orbiter in a window. You can specify the size of the render window here. Selecting one of the available *fixed aspect ratio* options (4:3 normal, 16:10 widescreen or 16:9 widescreen) automatically adjusts the window width or height to maintain the aspect ratio. Large window sizes can reduce simulation performance. Note that some older graphics drivers may not allow 3-D applications to run in window mode.

## 4.6 Joystick tab

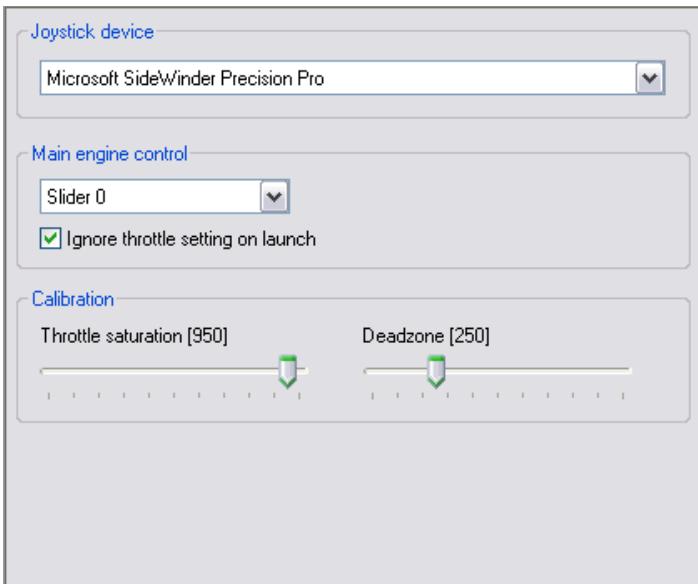
The *Joystick* tab allows selection and configuration of your joystick device, if present.

**Joystick device:** Lists all attached joysticks.

**Main engine control:** Define the joystick axis which controls the main thrusters. Try different options if the throttle control on your joystick doesn't work in Orbiter.

**Ignore throttle setting on launch:** If ticked, the joystick throttle will be ignored at the launch of a scenario until the user manipulates it. Otherwise, the throttle setting is used immediately.

**Deadzone:** Use this to define how soon the joystick will respond when moved out of its centre position. Smaller values make it respond sooner. Increase if attitude thrusters do not cut out completely in neutral position.



**Throttle saturation:** Defines the tolerance zone at the minimum and maximum range of the throttle control at which the joystick reports zero and maximum throttle,

respectively. Reduce if main engines do not cut out completely at minimum throttle setting. (Applies only to joysticks with throttle control).

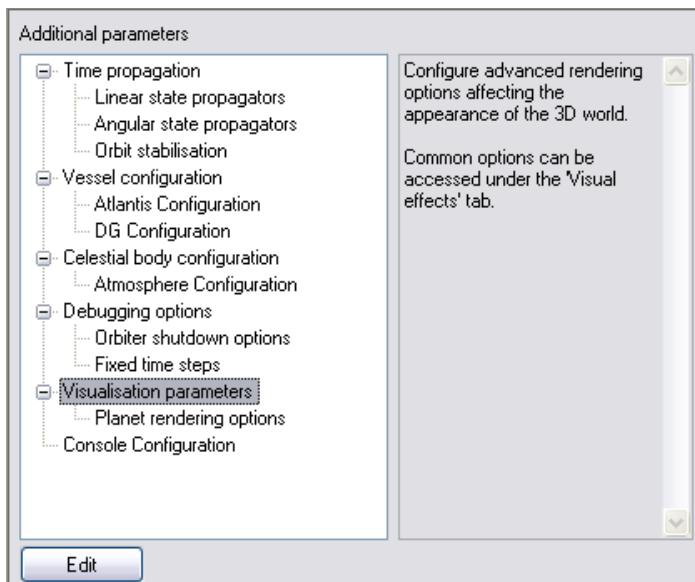
If further calibration is required you should use the appropriate tools in the Windows Control Panel.

## 4.7 Extra tab

The *Extra* tab contains a list of more advanced and specialised settings and configuration parameters, including details about Orbiter's dynamic state propagation, vessel configuration and debugging options. Addon plugins may add their own configuration entries to the list when activated.

It is generally safe for new users to leave all settings in this list at their default values. Advanced users can fine-tune the behaviour of the simulator here.

Click on an item to see a short description of its purpose to the right of the list. Double-clicking, or pressing the *Edit* button opens the associated configuration dialog. Among the configuration options available are:



**Time propagation** - defines the parameters for dynamic update of linear (position and velocity) and angular vessel states (orientation, angular velocity). Users can select the integration methods as a function of step interval. The Orbit stabilisation entry allows to configure the conditions under which Orbiter switches from dynamic to orbit perturbation updates. For technical details on the dynamic propagation schemes available in Orbiter, refer to the Orbiter Technical Note [Doc/Technotes/Dynamics](#).

**Vessel configuration** - Different spacecraft types may provide options for defining visual and physical behaviour under this section.

**Celestial body configuration** - Parameters to define particular characteristics of planetary bodies. Currently, this section contains configuration options for the atmospheric models of some planets.

**Debugging options** - Miscellaneous settings, including the way Orbiter shuts down a simulation session, and the option to enforce fixed time steps, which can be useful for debugging or trajectory generation.

**Visual parameters** - This section contains advanced rendering and texture load options for planetary bodies.

## 4.8 About Orbiter tab

The *About Orbiter* tab contains version and build information, as well as links to the Terms of Use, credits, and the Orbiter home page and forum.



ORBITER Space Flight Simulator  
Build 20 May 2010 [v.100520]  
Copyright (c) 2000-2010 Martin Schweiger  
[martins@medphys.ucl.ac.uk](mailto:martins@medphys.ucl.ac.uk)

Contact

Orbiter home: [orbit.medphys.ucl.ac.uk/](http://orbit.medphys.ucl.ac.uk/)  
Alias: [www.orbitersim.com](http://www.orbitersim.com)

Orbiter web forum: <http://orbiter-forum.com/>

Orbiter mailing list: [firedrake.org/orbiter/list.html](http://firedrake.org/orbiter/list.html)

Orbiter is free for personal and noncommercial use.  
Licensing details can be found under 'Terms of Use' below, or at the official  
Orbiter site.

[www.orbitersim.com](http://www.orbitersim.com) [Terms of Use](#) [Credits and Contributions](#)

## 5 Quickstart

This section demonstrates how to take off and land with one of Orbiter's default spacecraft, the Delta-glider. If you are using Orbiter for the first time, this will help to familiarise yourself with some basic concepts of spacecraft and camera control. You should also read the rest of this manual, in particular sections 6 and 8 on keyboard and joystick interface, section 14 on instrumentation, section 15 on spacecraft controls, and section 17 on basic flight maneuvers.

Make sure you have configured Orbiter before launching your first simulation, in particular the video and joystick parameters (see section 4). Once you have started the Quickstart scenario, you can get the following scenario instructions also on-screen by opening the Help window with **Alt F1**.

### Starting:

- ▶ Select the *Checklists|Quickstart* scenario (see Section 4.1 on scenario selection), and press the “Launch Orbiter” button to launch the scenario. Once the mission has been loaded (this can take a few moments), you will see in front of you runway 33 of the SLF (Shuttle Landing Facility) at the Kennedy Space Center, Cape Canaveral, Florida.
- ▶ You are in control of a Delta-glider, a powerful futuristic spacecraft, aligned and ready for takeoff.
- ▶ You can always exit the simulation by pressing **Ctrl Q** or **Alt F4**, or by clicking “Exit” on the main menu (**F4**). Orbiter saves the current simulation status in the “*(Current status*)” scenario, so you can continue your flight later by selecting this scenario.

### Camera modes:

You are in an external camera mode, looking towards your ship.

- ▶ You can rotate the camera around your ship by pressing and holding down the **Ctrl** key and pressing a cursor key (**↓↑→←**) on the cursor keypad of your keyboard. Alternatively you can press the right button on your mouse and drag the mouse to rotate the camera. Or, if you have a joystick with a direction controller (“coolie hat”), you can use that as well.
- ▶ To jump into the cockpit of your glider, press **F1**. (**F1** always toggles between cockpit and external view of the spacecraft you are controlling).
- ▶ In the cockpit, you can look around by rotating the camera with **Alt ↓↑→←**, or with the right mouse button or the joystick coolie hat.
- ▶ To look straight ahead, press the **Home** button.
- ▶ To learn more about camera modes and views, have a look at Section 12.

### Cockpit modes:

- ▶ At the moment, you are in "virtual cockpit" mode - that is, you are inside a three-dimensional representation of the glider cockpit, with the glass pane of the head-

up display (HUD) in front of you, and the instruments and controls arranged around you. If you look back, you can even get a glimpse of your passengers in the cabin behind you!

- ▶ You can switch to a different cockpit mode by pressing **[F8]**. Pressing **[F8]** once will open the "generic glass cockpit" mode with only the HUD and two onscreen multifunctional displays. Pressing **[F8]** again will open a 2-D panel mode.
- ▶ The panel can be scrolled by pressing a cursor key (**[↓][↑][→][←]**) on the cursor keypad. To scroll the panel out of the way, press **[↑]**. You should now be able to see the runway stretching in front of you. Scrolling the panel is useful if you want to see more of your surroundings. Also, if the panel is larger than your simulation window, you can scroll different parts of the panel into view.
- ▶ If the native resolution of the panel is larger than your simulation window, you can use the mouse wheel to zoom the panel view in and out (this feature may not be supported by all spacecraft types).
- ▶ Some spacecraft have more than a single panel which can be accessed by pressing **[Ctrl]** in combination with a cursor key. If you press **[Ctrl][↑]**, you will see the glider's overhead panel with some additional controls. Pressing **[Ctrl][↓]** twice will bring up the lower panel with brake and gear controls. For now, switch back to the main panel with **[Ctrl][↑]**.
- ▶ Not all spacecraft types support 2-D panels or 3-D virtual cockpits, but the generic cockpit mode is always available.

### MFD instruments:

The most important and versatile instruments are the two *multifunctional displays* (MFDs) in the centre of the instrument panel. Each MFD consists of a square LCD screen and buttons along the left, right and bottom edges.

- ▶ MFDs can be set to different modes: With the mouse, left-click the "SEL" button at the bottom edge of one of the MFDs. (Alternatively, you can press **[Shift][F1]**. MFD keyboard interfaces always use **[Shift]** key combinations, where the left **[Shift]** key controls the left MFD, and the right **[Shift]** key controls the right MFD). You will see a list of available modes.
- ▶ Click on one of the buttons along the left or right edge to select the corresponding mode. If you click the top-left button, the MFD switches to *Orbit* mode.
- ▶ If you want to select a mode via keyboard, press **[Shift][F1] + [Shift][letter]**, where **[letter]** is the keyboard character listed in grey next to the MFD mode in the selection page.
- ▶ Most modes have additional settings and parameters that can be controlled with the buttons as well. The button labels change to indicate the various mode functions. For example, the Orbit mode has a button labeled "TGT". This can be used to display the orbit of a target object. Click this button – you will see a dialog box to select a target object. Press **[Enter]**, type "iss" in the text box, and press **[Enter]**

again. This will show the orbital parameters of the International Space Station in the MFD display.

- ▶ To see a short description of the available mode functions, click the “MNU” button at the bottom of the MFD (Alternatively, use **Shift** **1**).
- ▶ A description of standard MFD modes can be found in Section 14. Orbiter can also be extended with add-on MFD modes, so you may see additional modes in the list.
- ▶ For now, switch the left MFD to *Surface* mode, and the right MFD to *HSI* mode.



### Takeoff:

Your glider is capable of runway takeoffs and landings on Earth (and on any other planet, if the atmospheric density is sufficient to provide aerodynamic lift).

- ▶ For takeoff, engage main engines at full thrust. You can do this by pushing the *Main* engine sliders at the left of the panel to the top using the mouse (make sure you push both sliders simultaneously!), or by pressing **Ctrl** **+** **Num** until engines are at full throttle. If you have a joystick with throttle control, you can use that to engage the main engines.
- ▶ Your spacecraft will start to roll. You can check the speed (in meters/second) on the AIRSPD indicator of the Surface MFD, or on the HUD (head-up display) – the value in the green box at the top right of the screen.
- ▶ When the airspeed reaches 100 m/s, pull back on the joystick to rotate, or press and hold **2** **Num**.

- Once clear of the runway, press **G** to raise the landing gear.



When the atmosphere is too thin to produce enough lift for a runway takeoff (for example when taking off from the Moon) or when no runway is available, you can use the glider's *hover* engines to lift off:

- Move the Hover slider on the instrument panel up by clicking and dragging with the mouse. Alternatively, press the **0<sub>Num</sub>** key until hover engines are fully engaged.
- Your glider should now lift off vertically. Once clear of the ground, engage main engines. Note that a fully loaded and tanked glider may be too heavy to lift off vertically from Earth when the "realistic" flight model is used.
- As you gain airspeed, you can gradually reduce hover thrust.

### **Atmospheric flight:**

In the lower atmosphere, the glider behaves very much like an aircraft. Try the joystick controls for pitch, roll and yaw to get a feeling for handling at different altitudes. Without a joystick, you can use the numerical keypad (**2**/**8<sub>Num</sub>** for pitch, **4**/**6<sub>Num</sub>** for roll, and **1**/**3<sub>Num</sub>** for yaw). The glider has powerful rocket engines, but their performance depends on atmospheric pressure (at very low altitudes, it will not even go supersonic).

This is a good time to try different camera modes. Open the Camera dialog (**Ctrl F1**), and check the effect of different track modes and field of view (FOV) settings.

### **Landing:**

- Go around and approach runway 33 of the SLF from the south. Line up with the runway. Your HSI instrument helps to maintain the correct approach path and slope. One of its two displays should already be tuned to the runway ILS system. The HSI contains a course pointer, deviation and glideslope indicator. It works

like a standard aircraft instrument, so you may already be familiar with its use. If not, check section 14.4 for details.

- ▶ As you approach the runway, you will see PAPI and VASI landing aids in front of and beside the runway (see section 17.6). The PAPI is of limited use here, because it is adjusted for the Space Shuttle's steep descent slope of 20°.
- ▶ Throttle back and engage airbrakes (**Ctrl** **B**) to reduce speed. Lower the landing gear (**G**).
- ▶ After touchdown, engage left and right wheel brakes (**L** and **R**) until you come to a full stop.

### **Space flight:**

So far we have treated the glider much like a conventional aircraft. Now it is time to aim a bit higher ...

- ▶ Take off as before. Turn east (use the compass ribbon at the top edge of the HUD, or the one in the Surface MFD display), and pitch up to 50°.
- ▶ As you gain altitude, you will notice that your craft starts to behave differently, due to the reduction in atmospheric pressure. One of the effects is a loss of lift, which causes the flight path indicator (the  $\oplus$  HUD marker) slowly to drift down. Another effect is the loss of response from your aerodynamic control surfaces.
- ▶ At about 30km altitude your glider will start to drop its nose even while you are pulling back on the stick. Now activate the RCS (Reaction Control System) by right-clicking the “RCS Mode” selector (on the right side of the instrument panel) or by pressing **Ctrl** **I** **Num**. You are now controlling your craft with attitude thrusters.
- ▶ Pitch down to about 20°. After leaving the dense part of the atmosphere, you need to gain tangential velocity to achieve orbit. Your flight path indicator should stay above 0°.
- ▶ Now is a good time to activate the Orbit mode in one of your MFDs. This shows the shape of your current orbit (the green curve) in relation to the planet surface (the gray circle), together with a list of orbital parameters along the left side of the display. You should switch the display to “current orbital plane” projection mode, by clicking on the “PRJ” button until “Prj: SHP” is shown in the top right corner of the display. Also select altitude readouts by clicking the “DST” button so that the PeR and ApR entries in the data column change to PeA and ApA (periapsis altitude and apoapsis altitude), respectively.
- ▶ At the moment, your orbit will be a rather eccentric ellipse, which for the most part is *below* Earth's surface. This means that you are still on a ballistic trajectory rather than in a stable orbit. As you keep gaining tangential velocity, the orbit will start to expand. Once the green curve is completely above the planet surface (and sufficiently high above the atmosphere) you will have entered orbit.
- ▶ At this point, the most important pieces of information from the Orbit display are the orbital velocity (“Vel”) and apoapsis altitude (“ApA”). For a low Earth orbit,

you need to achieve a velocity of at least 7800 m/s. Once you reach this value, you will see the orbit rising rapidly above Earth's surface. At the same time, the apoapsis altitude (the highest point of the orbit) will start to grow. Keep firing your engines until ApA reaches about +300km. Now cut the engines.

- ▶ You are now nearly in orbit. All that remains to do is raise the periapsis (the lowest point of the orbit) to a stable altitude. This is done best when you reach apoapsis, which should be half an orbit (or about 45 minutes) from your current position. Time to switch into an external camera mode and enjoy the view!
- ▶ It is also a good idea to switch the HUD from surface to orbit mode now. Do this by clicking the "OBT" button in the top left corner of the instrument panel, or by pressing **[H]** twice. In this mode, the HUD flight path ladder is aligned with the orbital plane instead of the horizon plane, and there is a ribbon showing your orbital azimuth angle. It also shows indicators for prograde (the direction of your orbital velocity vector) and retrograde (the opposite direction).
- ▶ When you approach apoapsis, turn your craft prograde. You can see how close you are to the apoapsis point by checking the ApT (time to apoapsis) value in the Orbit MFD. If it takes too long, press **[T]** to engage time acceleration, and **[R]** to switch back. To turn prograde, you can activate the RCS manually, but it is easier to leave it to the automatic attitude control, by simply pressing the "Prograde" button on the right of the instrument panel (or **[I]**).
- ▶ Now fire your main engines for final orbit insertion. The two parameters to watch are the orbit eccentricity ("Ecc") and periapsis altitude ("PeA"). The eccentricity value should get smaller, indicating that the orbit becomes more circular, while the periapsis altitude approaches the apoapsis altitude (ApA). Once the eccentricity value reaches a minimum, turn the main engines off. You can also deactivate the prograde attitude mode by clicking "Prograde" again.
- ▶ Congratulations! You made it into orbit!

### **Deorbiting:**

Should you ever want to come back to Earth, you need to *deorbit*. This means to drop the periapsis point to an altitude where the orbit intersects the dense part of the atmosphere, so that your vessel is slowed down by atmospheric friction.

- ▶ Deorbit burns are performed retrograde. Click the "Retrograde" button, wait until the vessel attitude has stabilised, and engage main engines.
- ▶ Keep burning until the periapsis point is well below Earth's surface ( $PeA < 0$ ), then cut the engines. Strictly speaking, the deorbit burn must be timed precisely, because too shallow a reentry angle will cause you to skid off the atmosphere, while too steep an angle will turn you into a shooting star. For now we are not concerned with such fine detail...
- ▶ Turn prograde again and wait for your altitude to drop. As you enter the lower part of the atmosphere, friction will cause your velocity to decrease rapidly. Reen-

tries are usually performed with a high angle of attack (AOA) – about  $40^{\circ}$  for the Space Shuttle.

- ▶ Once your aerodynamic control surfaces become responsive again you can turn off the RCS system. Your glider has now turned back into an aircraft.
- ▶ You have probably ended up a long way from your launch point at the KSC. Re-entering towards a specified landing point requires some practice in timing the deorbit burn and the reentry flight path. We'll leave this for a later mission. For now, simply look for a dry patch to land your glider.
- ▶ This completes your first orbital excursion!

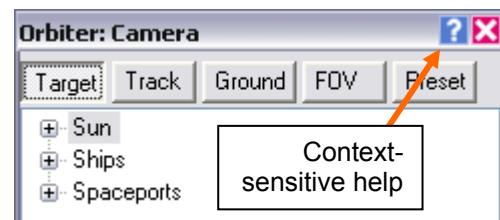
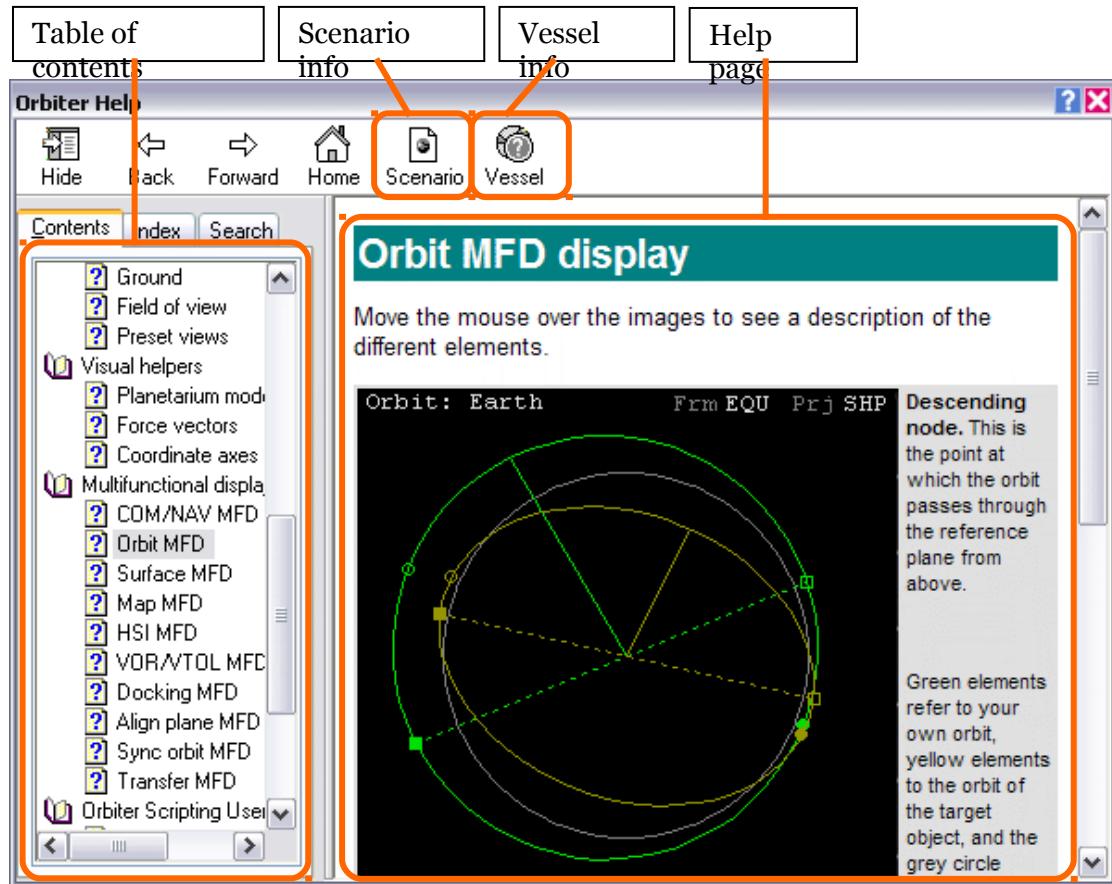
You are now ready to try more advanced missions. Try the “Launch to docking with the ISS” flight described in section 21. First you might want to learn a bit more about orbital maneuvers and docking procedures in section 17.

## 6 The help system

From the Orbiter Launchpad, you can get a description of the different dialog box options by pressing the “Help” button in the bottom right corner.

During the simulation you can open the Orbiter help window by pressing **[Alt] [F1]**, or by selecting “Help” from the main menu (**[F4]**).

The help system provides information about MFD modes, and optionally a description of the current scenario or the currently active spacecraft.



The help system is currently still under development. Not all scenarios and vessels currently support context-sensitive help. The system can be extended by adding additional scenario and vessel help pages, and add-on developers are encouraged to use the help system to provide user-friendly information about their spacecraft, or to include documented tutorial scenarios that illustrate the features of their plug-ins.

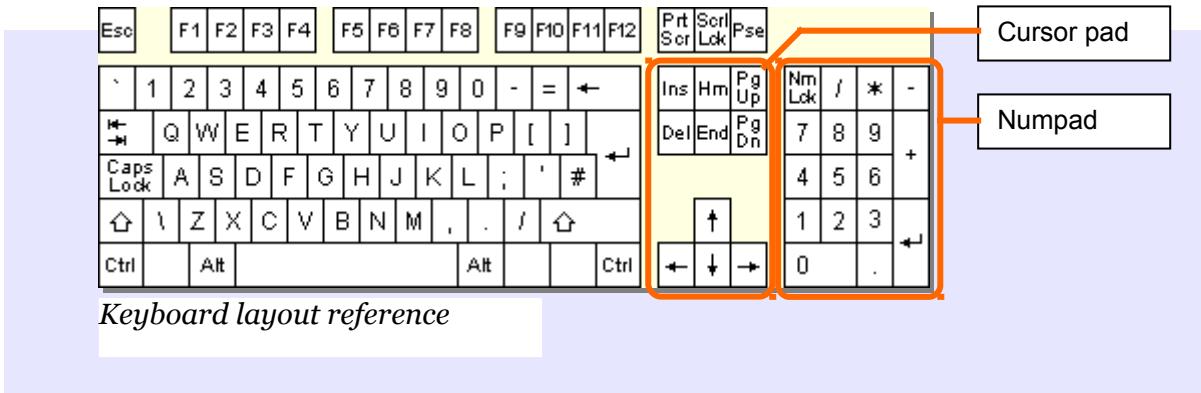
## 7 Keyboard interface

This section describes the *default* Orbiter keyboard functions. Please note that the key assignments are customizable by editing the keymap.dat file in the orbiter directory, and that therefore the keyboard controls for your Orbiter installation may be different.

The key assignment reference in this section and the rest of the manual refers to the keyboard layout shown in the figure below. For other layouts (e.g. language-specific) the key labels may be different. The relevant criterion for key functions in Orbiter is the *position of the key* on the keyboard, not the key label. For example, on the German keyboard, the keys for the “turn orbit-normal” (;) and “turn orbit-antinormal” (‘) will be “Ö” and “Ä”.

Keys from the numerical keypad or the cursor keypad will be denoted by subscript, e.g.  $\square_{Num}$  or  $\square_{Cur}$ .

Note that certain spacecraft may define additional keyboard functions. Check individual manuals for a detailed description of spacecraft controls and functionality.



	Time warp shortcut: Slow down simulation by factor 10 (down to real-time). See also Time acceleration dialog ( $\square_{Ctrl} \square_{F2}$ )
	Time warp shortcut: Speed up simulation by factor 10 (up to a maximum warp factor of 100000). See also Time acceleration dialog ( $\square_{Ctrl} \square_{F2}$ )
	Zoom out (increase field of view). See also Camera dialog ( $\square_{Ctrl} \square_{F1}$ )
	Zoom in (decrease field of view). See also Camera dialog ( $\square_{Ctrl} \square_{F1}$ )
	Zoom out (in discrete steps of 10°).
	Zoom in (in discrete steps of 10°).
	Start/stop recording a flight, or stop a flight playback. See also Flight recorder dialog ( $\square_{Ctrl} \square_{F5}$ )
	Undock from a vessel.
	Pause/resume simulation.
	Exit to Launchpad dialog.
	Quicksave scenario.

	Toggle internal/external view of the user-controlled spacecraft.
	Open the <i>Camera</i> dialog to select camera target, view mode and field of view.
	Opens the online help window.
	Toggle tracking mode for external camera views (target-relative / absolute direction / global frame).
	Open the <i>Time acceleration</i> dialog. This allows to speed up/slow down the simulation, and to pause/resume.
	Open vessel dialog to switch control to a different spacecraft.
	Switch control back to the previously active vessel. This allows quickly switching backwards and forwards between two vessels.
	Main menu.
	Open the <i>Custom functions</i> dialog. Contains a list of functions defined in plug-in modules, if available.
	Open the <i>Flight recorder/playback</i> dialog. Contains recording and playback options.
	Open the <i>Object Info</i> dialog for object-specific data such as ILS navaid frequencies etc.
	Open the <i>Map</i> dialog (spaceports, navaid locations etc.)
	Open the <i>Navaid Info</i> dialog containing a list of navigational radio beacons.
	Open the <i>Planetarium options</i> dialog for controlling the display of grids and markers.
	“Planetarium mode”: Toggle display of constellations.

## 7.2 Spacecraft controls

These keys allow manual maneuvering of the user-controlled spacecraft. See also joystick controls. Note some spacecraft may not define all thruster types.

### Main/retro thruster controls:

	Accelerate by increasing main thruster setting or by decreasing retro thruster setting.
	Decelerate by decreasing main thruster setting or by increasing retro thruster setting.
	Kill main and retro thrusters.
	Fire main thrusters at 100% while pressed (overrides permanent setting)
	Fire retro thrusters at 100% while pressed (overrides permanent setting)

### Hover thruster controls (where available):

	Increase hover thruster setting.
	Decrease hover thruster setting.

### Attitude thruster controls (rotational mode):

	Engage attitude thrusters for rotation around longitudinal axis (bank)
	Engage attitude thrusters for rotation around transversal axis (pitch)
	Rotational mode: Engage attitude thrusters for rotation around vertical axis

	(yaw)
<b>5</b> Num	Toggle “Kill rotation” navigation computer mode. Stops spacecraft rotation by engaging appropriate attitude thrusters

**Note:** In combination with , thrusters are engaged at 10% max. thrust for fine control.

#### Attitude thruster controls (linear mode):

<b>2</b> / <b>8</b> Num	Engage attitude thrusters for up/down translation.
<b>1</b> / <b>3</b> Num	Engage attitude thrusters for left/right translation.
<b>6</b> / <b>9</b> Num	Engage attitude thrusters for forward/back translation

**Note:** In combination with , thrusters are engaged at 10% max. thrust for fine control.

#### Other controls:

<b>1</b> Num	Toggle reaction control thruster mode between rotational (engage opposite thruster pairs) and linear (engage parallel thruster pairs)
<b>1</b> Num	Enable/disable reaction control system (RCS). The RCS (if available) is a set of small thrusters which allows attitude (rotation) and linear control of the spacecraft.
<b>1</b> Num	Enable/disable manual user control (via keyboard or joystick) of aerodynamic control surfaces (elevator, rudder, ailerons) if available.
<b>A</b>	Toggle “Hold altitude” navcomp mode. Maintain current altitude above surface by means of hover thrusters only. This will fail if hover thrusters cannot compensate for gravitation, in particular at high bank angles. Combining this mode with the “H-level” mode is therefore useful.
<b>L</b>	Toggle “H-level” navcomp mode. This mode keeps the spacecraft level with the horizon by engaging appropriate attitude thrusters.
<b>I</b>	Toggle “Turn prograde” navcomp mode. This mode turns the spacecraft into its orbital velocity vector.
<b>J</b>	Toggle “Turn retrograde” navcomp mode. This mode turns the spacecraft into its negative orbital velocity vector.
;	Toggle “Turn orbit-normal” navcomp mode. Rotates spacecraft normal to its orbital plane (in the direction of $\vec{R} \times \vec{V}$ )
,	Toggle “Turn orbit-antinormal” navcomp mode. Rotates spacecraft anti-normal to its orbital plane (in the direction of $-\vec{R} \times \vec{V}$ )
<b>Insert</b> / <b>Delete</b> Cur	Trim control (only vessels with aerodynamic surfaces)
,	Apply left wheel brake (where applicable)
.	Apply right wheel brake (where applicable)

### 7.3 External camera views

<b>Page Up</b>	Move camera away from target object.
<b>Page Down</b>	Move camera towards target object.
	Rotate camera around object.
<b>↑</b> <b>↓</b> <b>→</b> <b>←</b>	

In ground-based camera views, **↑** **↓** **→** **←** will move the observer position, **Page Up** and **Page Down** will change the observer altitude, and **↑** **↓** **→** **←** will rotate the observer direction (unless locked to the target).

## 7.4 Internal (cockpit) view

The two multifunctional displays (MFD) on the left and right side of the screen are controlled via left/right Shift key combinations, where the left Shift key addresses the left MFD, the right shift key addresses the right MFD.

The Head-up display (HUD) and MFDs are visible only in internal cockpit view.

	Toggle between generic, 2D panel, and 3D virtual cockpit modes (if supported by the current spacecraft)
	Rotate view direction.
	Return to default view direction.
	Scroll instrument panel (in 2D panel view).
	Switch to neighbour panel, if available (in 2D panel view).
	Toggle HUD display on/off.
	Toggle HUD colour.
	Switch HUD mode.
	HUD reference selection. Orbit HUD: opens reference selection input box. Docking HUD: steps through available NAV receivers.
	Docking HUD: Reference selection, bypassing XPDR and IDS transmitters.

## 7.5 MFD control

MFD commands are generally -key commands, where the left and right keys refers to the left and right MFD display, respectively.

	Toggle MFD on/off (equivalent to MFD PWR button).
	Open a menu for MFD mode selection (equivalent to MFD SEL button).
	Open/page/close the MFD-specific parameter selection menu (equivalent to MFD MNU button).
	In MFD-mode selection,  in combination with a mode key selects that mode (see Section 14).
	In standard display mode,  in combination with a mode-specific function key activates that function (see Section 14).

For control of specific multifunctional display (MFD) modes see Section 14 or the quick reference in 24.

## 7.6 Menu selections

	Move to previous item in the list.
	Move to next item in the list.
	Display sub-list for selected item, if available.
	Go back to the parent list from a sub-list.

 Enter	Select current item and close list.
 Esc	Cancel list.

## 8 Joystick interface

A joystick can be used to operate the attitude and main thrusters of the user-controlled spacecraft manually.

Action	Effect
Push stick left or right	Rotate around vessel's longitudinal axis (bank)
Push stick forward or backward	Rotate around vessel's transversal axis (pitch)
Operate rudder control <i>or</i> Push stick left or right while holding joystick button 2	Rotate around vessel's vertical axis (yaw)
Operate throttle control	Controls main thruster settings. This is similar to the $\text{Ctrl} + \text{Num}$ and $\text{Ctrl} - \text{Num}$ keyboard controls, but it affects only the main thrusters, not the retro thrusters.
Direction controller ("coolie hat")	Cockpit view: rotate view direction External view: rotate camera around the observed object
Direction controller + joystick button 2	Cockpit view: scroll instrument panel (if applicable) External view: rotate view direction (ground observer mode only)

## 9 Mouse interface

Spacecraft instrument panels can be operated by the mouse. Most buttons, switches and dials are activated by pressing the left mouse button. Some elements like multi-way dials may respond to both left and right mouse buttons. In generic cockpit view, the buttons around the two multifunctional displays (MFDs) can be operated with the mouse.

In external camera modes, the *mouse wheel control* (if available) can be used to move the camera towards or away from the view target. The mouse wheel acts like the  and  keys.

In internal 2-D panel cockpit views, the mouse wheel can be used to zoom the panel view in and out, if the native resolution of the panel is higher than the size of the simulation window (not supported for legacy panel implementations).

The camera direction can be rotated by holding down the right mouse button and dragging the mouse. This works both in external and cockpit views.

The mouse can of course also be used to select and manipulate dialog controls.

## 10 Spacecraft classes

The following standard spacecraft types are currently available in the Orbiter standard distribution. Many more can be downloaded as add-ons. See the Orbiter web site for a list of add-on repositories.

### 10.1 Delta-glider

The Delta-glider (DG) is the ideal ship for the novice pilot to get space borne. Its futuristic design concept, high thrust and extremely low fuel consumption make it easy to achieve orbit, and it can even be used for interplanetary travel. The winged design provides aircraft-like handling in the lower atmosphere, while the vertically mounted hover-thrusters allow vertical takeoffs and landings independent of atmospheric conditions and runways.



*Delta-glider  
model and  
textures by Roger  
“Frying Tiger”  
Long. Instrument  
panels by Martin  
Schweiger.*

The DG supports 2-D instrument panels and a virtual cockpit in addition to the standard “glass cockpit” camera mode.

The glider comes with operating landing gear, nose cone docking port, airlock door, deployable radiator and animated aerodynamic control surfaces. It supports particle exhaust effects.

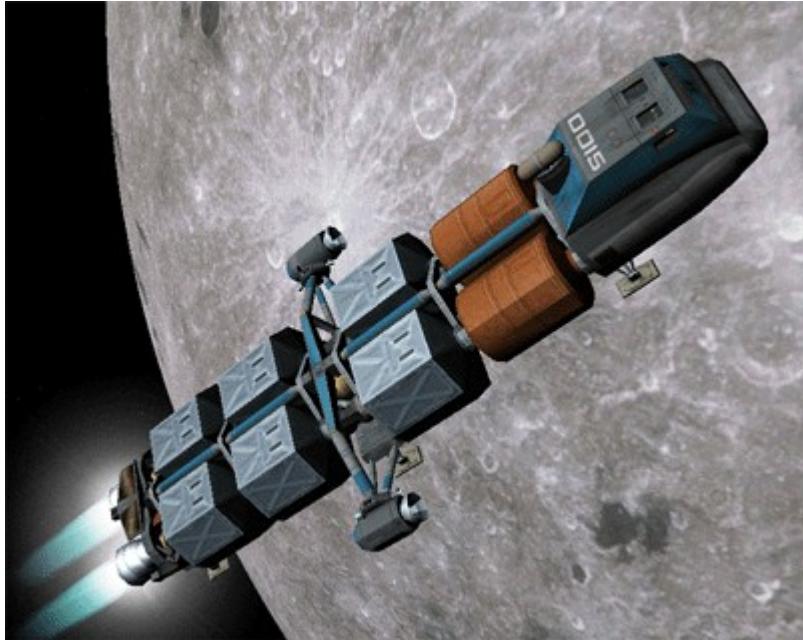
Details on instrumentation, controls, camera modes and technical specifications can be found in the separate document [Doc\DeltaGlider](#).

### 10.2 Shuttle-A

The Shuttle-A, designed by Roger “Frying Tiger” Long, is medium size freight-vessel, designed preliminary for low gravity/low density environments. The current design

allows to achieve LEO from Earth's surface, but you need to plan your ascent carefully not to run out of fuel.

The vessel has a set of two main and two hover thrusters, plus a pair of auxiliary thruster pods which can be rotated 180° for main, hover or retro thrust.



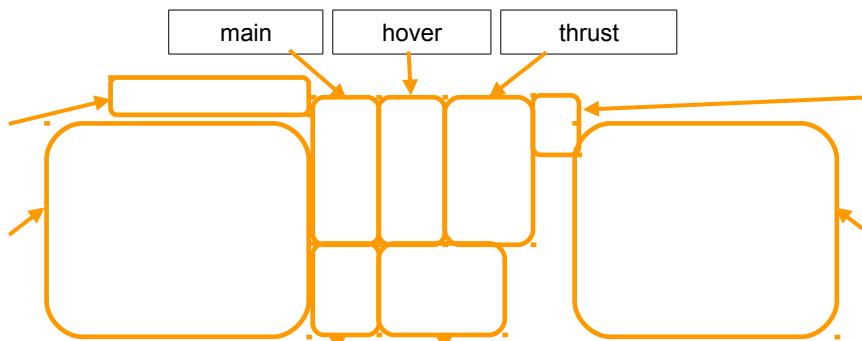
*Model design: Roger Long. Instrument panels and module code: Martin Schweiger. Virtual cockpit, payload management extensions: Radu Poenaru.*

The Shuttle-A comes with instrument panels. For operational details and technical specifications see the separate Shuttle-A Technical Manual.

The latest version of the Shuttle-A supports a virtual cockpit, detachable cargo pods, and working landing gear, contributed by Radu Poenaru.

### Main and overhead instrument panels:

Turn the panels on and off with **[F8]**. The Shuttle-A supports two panels which can be selected with **[Ctrl] [↓]** and **[Ctrl] [↑]**.



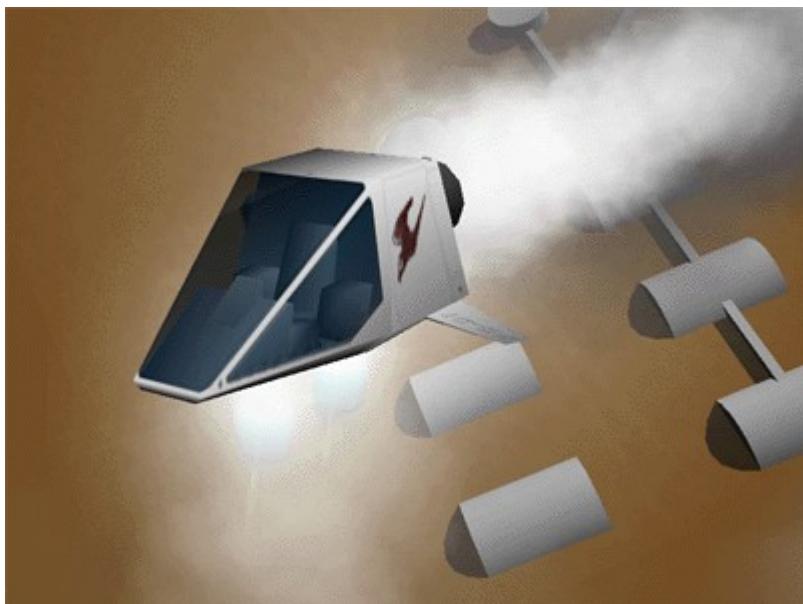


### Vessel-specific key controls:

	Operate docking hatch mechanism
	Open/close outer airlock door
	Operate landing gear.

## 10.3 Shuttle PB (PTV)

The PB is a very agile single-seater. It produces little lift in atmospheric flight, and depends on its hover thrusters for takeoff and landing. Aerodynamic control surfaces are not supported in this version. Attitude control is performed via the RCS (reaction control system).



*Overall design and textures: Balázs Patyi. Model improvements: Martin Schweiger*

### Technical specifications:

Mass	500 kg (empty orbiter)
	750 kg (fuel capacity)
	1250 kg (total)
Length	7 m
Thrust	$3.0 \cdot 10^4$ N (main)
	$2 \times 0.75 \cdot 10^4$ N (hover)
Isp	$5.0 \cdot 10^4$ m/s (fuel-specific impulse in vacuum)

## 10.4 Dragonfly

The Dragonfly is a space tug designed for moving payload in orbit. It may be used to bring satellites delivered by the Space Shuttle into higher orbits, or to help in the assembly of large orbital structures.

The Dragonfly has no dedicated main thrusters, but a versatile and adjustable reaction control system.

THE DRAGONFLY IS NOT DESIGNED FOR ATMOSPHERIC DESCENT OR SURFACE LANDING!



*Dragonfly original design: Martin Schweiger. Model improvements and textures: Roger Long. Systems simulation and instrument panels: Radu Poenaru.*

The Dragonfly is the first vessel to be modeled with detailed electrical and environmental systems simulation, contributed by Radu Poenaru. For detailed information see the Dragonfly Operations Handbook.

### Technical specifications:

Mass	$7.0 \cdot 10^3$ kg	(empty)
	$11.0 \cdot 10^3$ kg	(100% fuel)
Length	14.8 m	
Width	7.2 m	
Height	5.6 m	
<b>Propulsion system</b>		
RCS mounted in 3 pods (left, right, aft) total 16 thrusters		
Thrust rating	1.0 kN	per thruster
Isp	$4.0 \cdot 10^4$ m/s	(vacuum)

## 10.5 Space Shuttle Atlantis

Space Shuttle Atlantis represents the only “real” spacecraft in the basic Orbiter distribution (but there are many more available as addons). Its flight characteristics are less forgiving than fictional models like the Delta-glider, and just reaching orbit is a challenge.

The Atlantis orbiter features a working payload bay with remote manipulator system (“Canadarm”), so you can simulate the deployment or even recapture of satellites, or the shipment of resupplies to the International Space Station.

The model now also features a virtual cockpit, with working MFD instruments and head-up display, a working payload bay and remote manipulator arm, as well as MMU support.

Operation procedures and implementation details are provided in separate documents: [Doc\Atlantis](#) and [Doc\Atlantis\\_MMU\\_Sat\\_30](#).

Below are a few simplified checklists for launch, docking and payload operation.

**Launch:**

- ▶ Fire main engines at 100%.
- ▶ SRBs are ignited automatically when main engines reach 95%. SRBs are not controlled manually. Once ignited, they cannot be shut off.

*3D model and textures: Michael Grosberg, Don Gallagher (orbiter) and Damir Guleşich (ET+SRB). Original module code: Martin Schweiger. Original grappling, RMS and MMU extensions: Robert Conley. Module code extensions: David Hopkins and Douglas Beachy.*

*The virtual cockpit from the commander's seat.*

- ▶ SRBs separate automatically at T+2:06min. In an emergency, SRBs can be jettisoned manually with **[J]**.
- ▶ Ascent continues with Orbiter main engines. Throttle down as required for 3g max acceleration.
- ▶ Tank separates at T+8:58min (alt 110km) when empty, or manually with **[J]**.
- ▶ After tank separation, orbiter switches to OMS (orbital maneuvering system) using internal tanks, for final orbit insertion. Attitude thrusters (RCS – reaction control system) are activated.

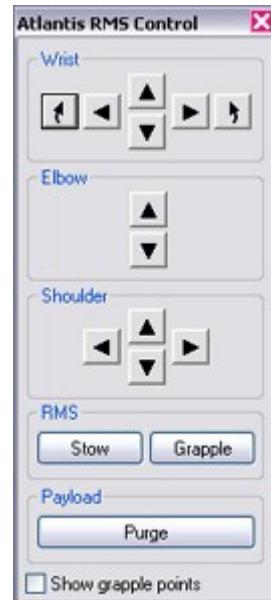
#### **Docking:**

- ▶ The orbiter carries a docking attachment in the cargo bay.

- ▶ Open cargo bay doors before docking.
- ▶ Docking direction is in orbiter's +y direction (up). The Docking MFD must be interpreted accordingly.

### RMS manipulation and grappling:

- ▶ The shuttle carries a mechanical manipulator arm in the cargo bay which can be used for releasing and recapturing satellites, MMU control, etc.
- ▶ The arm can be used in orbit once the cargo doors have been fully opened.
- ▶ To bring up the RMS control dialog, press **Ctrl**+Space.
- ▶ The arm has three joints: the shoulder joint can be rotated in yaw and pitch, the elbow joint can be rotated in pitch, and the wrist joint can be rotated in pitch, yaw and roll.
- ▶ To grapple a satellite currently stowed in the cargo bay, move the RMS tip onto a grappling point, and press "Grapple". If grappling was successful, the button label switches to "Release".
- ▶ To make it easier to identify the grappling points of satellites, you can tick the "Show grapple points" box. This marks all grappling points with flashing arrows.
- ▶ To release the satellite, press "Release".
- ▶ You can also grapple freely drifting satellites if you move the RMS tip onto a grappling point.
- ▶ To return a satellite back to Earth, it must be stowed in the cargo bay. Use the RMS arm to bring the satellite into its correct position in the payload bay. When the Payload "Arrest" button becomes active, the satellite can be fixed in the bay by pressing the button. It is automatically released from the RMS tip.
- ▶ The RMS arm can be stowed in its transport position by pressing the RMS "Stow" button. This is only possible as long as no object is attached to the arm.
- ▶ Payload can be released directly from the bay by pressing the "Purge" button.



### Atlantis-specific key controls:

[J]	Jettison: separate SRBs or main tank
[K]	Operate cargo bay doors. The cargo bay doors cannot be closed when the Ku-band antenna is deployed.
[G]	Operate landing gear (activated only after tank separation)
[Ctrl] [B]	Operate split-rudder speed brake.
[Ctrl] [U]	Deploy/retract Ku-band antenna. The antenna can only be operated if the cargo bay doors are fully open.
[Ctrl] + Space	Open RMS control dialog.

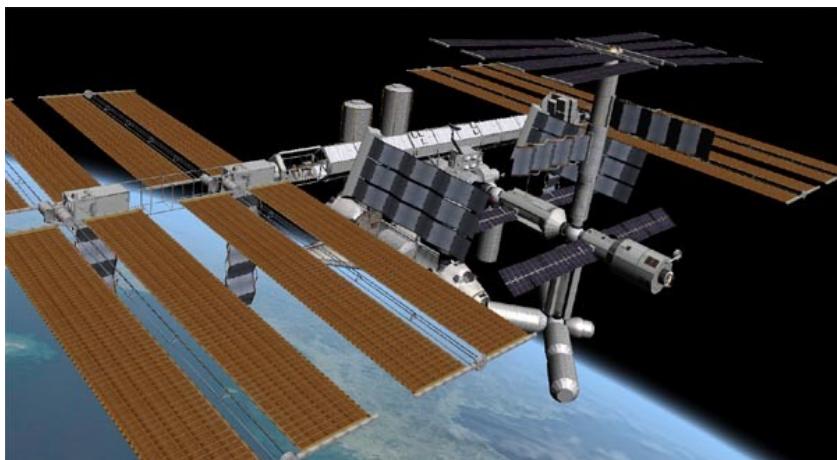


Unlike the futuristic spacecraft designs, Atlantis provides only a small margin of error for achieving orbit. Try some of the other ships before attempting to launch the Shuttle. Limited fuel *must* be selected, otherwise Atlantis is too heavy to reach orbit!

## 10.6 International Space Station (ISS)

The International Space Station is a multinational scientific orbital platform currently under construction (although its fate is now somewhat in doubt after the Columbia disaster).

Orbiter contains the ISS in its completed state. The ISS is a good docking target for Shuttle and other spacecraft missions.



The ISS contains 5 docking ports. In Orbiter, each is equipped with an IDS (Instrument Docking System) transmitter. The default IDS frequencies are:

Port 1	137.40
Port 2	137.30
Port 3	137.20
Port 4	137.10

Port 5      137.00

For docking procedures see Section 17.7.

## 10.7 Space Station MIR

In Orbiter, the Russian MIR station is still in orbit around Earth and can be used for docking approaches. Furthermore, unlike its real-life counterpart, Orbiter's MIR is orbiting in the plane of the ecliptic, which makes it an ideal platform to launch lunar and interplanetary missions.



*MIR model and textures by Jason Benson*

MIR sends a transponder (XPDR) signal at default frequency 132.10 which can be used for tracking the station during a rendezvous maneuver.

MIR supports 3 docking ports, with the following IDS transmitter frequencies:

Port 1      135.00

Port 2      135.10

Port 3      135.20

## 10.8 Lunar Wheel Station

This is a large fictional space station in orbit around the Moon. It consists of a wheel, attached to a central hub with two spokes. The wheel has a diameter of 500 metres and is spinning at a frequency of one cycle per 36 seconds, providing its occupants with a centrifugal acceleration of  $7.6 \text{ m/s}^2$ , or about  $0.8g$ , to mimic Earth's surface gravitational force.

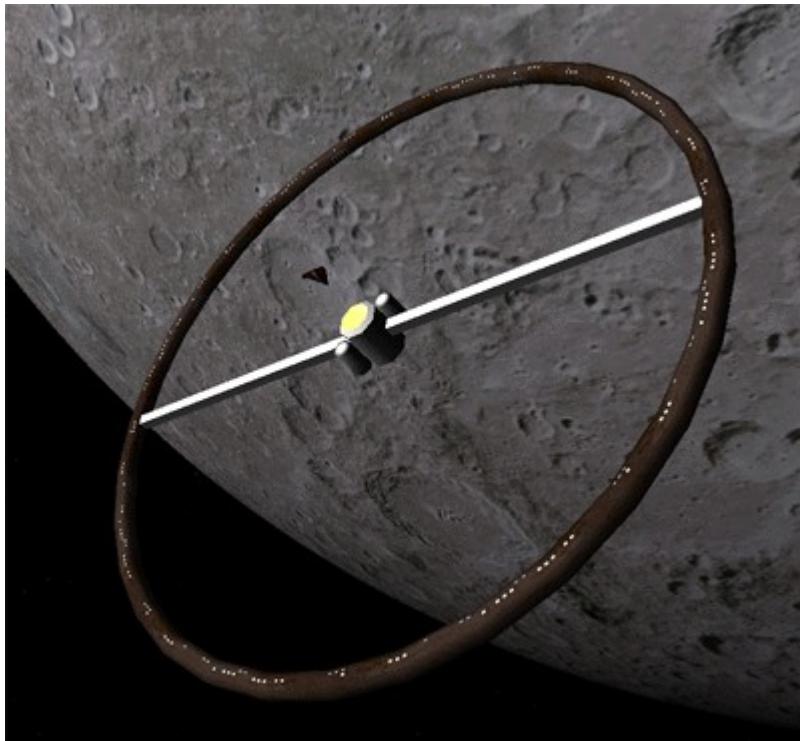
The main problem the station poses to the spacecraft pilot is in performing a docking maneuver. Docking to a rotating object is only possible along the rotation axis. The wheel has two docking ports in the central hub. The docking approach is performed along the axis of rotation. Before docking, the approaching vessel must synchronise its own longitudinal rotation with that of the station. For docking procedures, see Section 17.7.



Currently, Orbiter's docking instrumentation works on rotating docking targets *only* if the vessel's docking port is aligned with its longitudinal axis of rotation. This is the case for Shuttle-A and Dragonfly, but *not* for the Delta-glider or Space Shuttle.

The wheel station sends a transponder signal at frequency 132.70. The default IDS transmitter frequencies for the two docking ports are

Port 1	136.00
Port 2	136.20



*Wheel model and textures: Martin Schweiger*

## 10.9 Hubble Space Telescope

The Hubble Space Telescope is the visible/ultraviolet/near-infrared element of the Great Observatories astronomical program. The spacecraft provides an order of magnitude better resolution than is capable from ground-based telescopes. The objectives of the HST are to: (1) investigate the composition, physical characteristics, and dynamics of celestial bodies; (2) examine the formation, structure, and evolution of stars and galaxies; (3) study the history and evolution of the universe; and (4) provide a long-term space-based research facility for optical astronomy. During initial on-orbit checkout of the Hubble's systems, a flaw in the telescope's main reflective mirror was found that prevented perfect focus of the incoming light. This flaw was caused by the incorrect adjustment of a testing device used in building the mirror. Fortunately, however, Hubble was designed for regular on-orbit maintenance by Shuttle missions. The first servicing mission, STS-61 in December 1993, fully corrected the problem by installing a corrective optics package and upgraded instruments (as well as replacing other satellite components). A second servicing mission, scheduled for March 1997, installed two new instruments in the observatory.

Orbiter provides several Space Shuttle/HST missions for both deployment and re-capture operations. For Shuttle payload manipulation, see Section 10.5 above.

### HST-specific key controls:

 1	Deploy/retract high-gain antennae
 2	Open/close telescope tube hatch
 3	Deploy/fold solar arrays

## 10.10 LDEF Satellite

### Long Duration Exposure Facility (LDEF)

Deployed in orbit on April 7, 1984 by Shuttle Challenger and intended for retrieval after one year, the LDEF satellite was stranded in orbit for six years after the Challenger accident. The crew of STS-32 recovered the LDEF from its decaying orbit on Jan-



*HST model and textures by David Sundstrom*

uary 11, 1990, two months before it would have re-entered the Earth's atmosphere and would have been destroyed.

The LDEF makes a good object for deployment and retrieval missions in Orbiter.



*LDEF mesh by  
Don Gallagher.*

## 11 Object information

Use the object information window to retrieve data and current parameters about

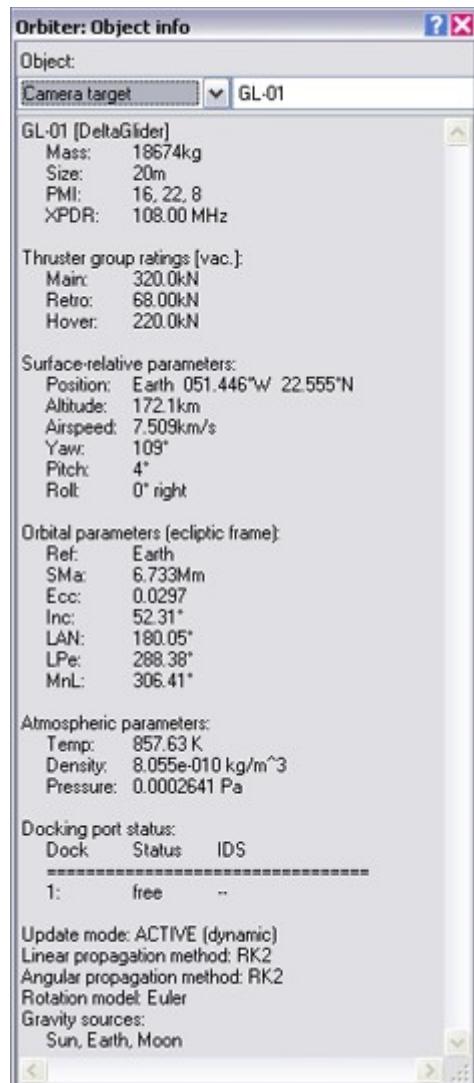
- ▶ the current camera target object
- ▶ spacecraft
- ▶ spaceports
- ▶ celestial objects (sun, planets, moons)

The object information window can be opened during the simulation by selecting *Object info* from the main menu, or by pressing **Ctrl I**.

### 11.1 Vessel information

Select object type *Vessel*, and pick one of the spacecraft in the current simulation from the list. The information sheet for spacecraft and orbital stations contains:

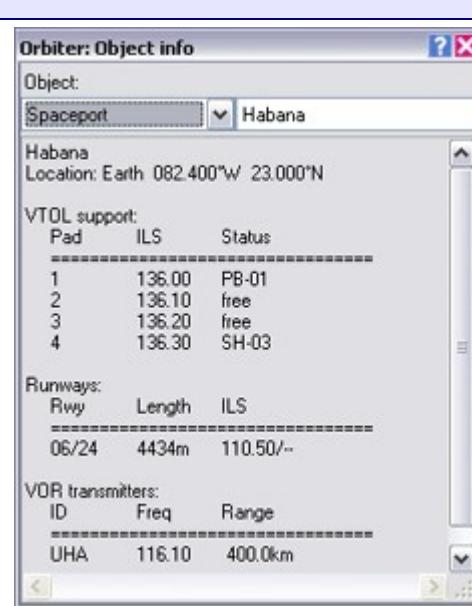
- ▶ current mass and size
- ▶ mass-normalised principal moments of inertia (PMI)
- ▶ transponder frequency
- ▶ engine vacuum thrust ratings
- ▶ equatorial position (longitude and latitude) above currently orbited planet, altitude and speed
- ▶ attitude relative to local horizon (yaw, pitch, roll angles)
- ▶ orbital elements in the ecliptic frame of reference, relative to currently orbited planet (semi-major axis, eccentricity, inclination, longitude of ascending node, longitude of periapsis, mean longitude at epoch)
- ▶ Atmospheric temperature, density and pressure
- ▶ docking port status, if applicable (free/docked vessel, instrument docking system [IDS] transmitter frequency)
- ▶ time propagation mode (free-flight/landed, dynamic or stabilised time step updates), and current gravitational field sources



## 11.2 Spaceport information

Select object type *Spaceport* and pick one of the available surface bases from the list. Spaceport information sheets contain:

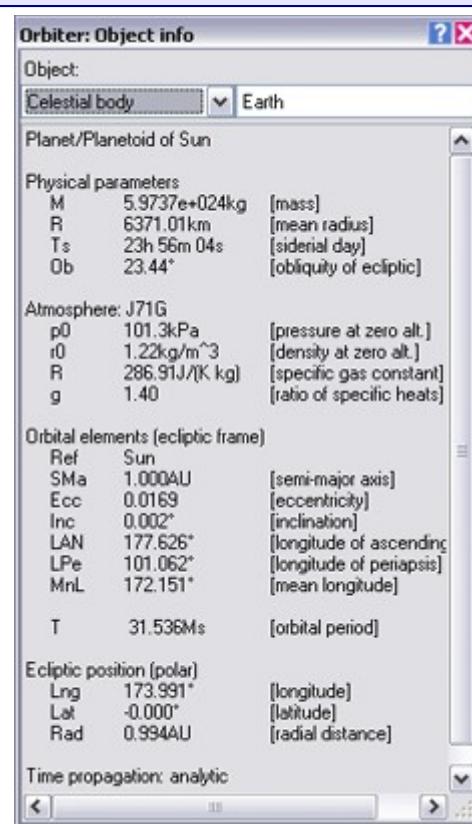
- ▶ planet/moon and equatorial position (longitude and latitude)
- ▶ landing pad status (free/landed vessel, and instrument landing system [ILS] transmitter frequency)
- ▶ runway information (runway alignment direction, length, and ILS transmitter frequency)
- ▶ frequencies and ranges for any VOR (very high frequency omnidirectional radio) transmitters associated with the spaceport.



## 11.3 Celestial body information

Select object type *Celestial body* and pick one of the bodies listed. Information sheets for celestial bodies (such as sun, planets and moons) contain:

- ▶ physical parameters:
  - ▶ mass (M)
  - ▶ mean radius (R)
  - ▶ length of siderial ("star") day (Ts)
  - ▶ obliquity of ecliptic (Ob) – tilt of axis of rotation against plane of ecliptic
- ▶ atmospheric parameters (if applicable):
  - ▶ atmospheric pressure at zero altitude (po)
  - ▶ atmospheric density at zero altitude (ro)
  - ▶ specific gas constant (R)
  - ▶ ratio of specific heats cp/cv (g)
- ▶ orbital elements in the ecliptic frame of reference, relative to currently orbited body (semi-major axis, eccentricity, inclination, longitude of ascending node, longitude of periapsis, mean longitude at epoch)
- ▶ current ecliptic position in polar coordinates (longitude, latitude and radius) relative to currently orbited body.



- ▶ geocentric celestial position (right ascension and declination)

## 12 Camera modes

Orbiter's solar system contains a variety of objects, including planets, moons, space-craft and launch sites. You can have a look at any of these by adjusting the camera mode. To open the camera configuration dialog, press **Ctrl F1**. You can now

- ▶ Point the camera to a new target, by selecting an object from the list, and clicking *Apply*.
- ▶ Jump back to the current focus object in external or cockpit view, by clicking *Focus Cockpit* or *Focus Extern*. (Shortcut: **F1**)
- ▶ Select the external camera tracking or ground-based mode, by clicking the *Track* tab. (Shortcut: **F2**)
- ▶ Change the camera field of view, by clicking the *FOV* tab. (Shortcut: **Z** and **X** for continuous zooming, and **Ctrl Z** and **Ctrl X** for discrete zoom steps).
- ▶ Store and recall camera modes via the preset list, by clicking the *Preset* tab.



### 12.1 Internal view

In internal (cockpit) view the player is placed inside the cockpit of his/her spaceship and looks forward. Instrument panels, head-up display (HUD) and multifunctional displays (MFD) are only shown in internal view. To return to cockpit view from any external views, press **F1**, or select *Focus Cockpit* from the camera dialog.

Some spacecraft types support scrollable 2D instrument panels and/or a 3-dimensional “virtual cockpit”, in addition to the generic view. Press **F8** to switch between the available cockpit modes.

You can rotate the view direction by pressing the **Alt** key in combination with a cursor key (**↓↑→←**) on the cursor keypad. To return to the default view direction, press **Home** on the cursor keypad.



2D panels can be scrolled with **↓↑→←**. This is useful if the panel is larger than the simulation window, or to scroll the panel out of the way.

If a ship supports multiple panels, you can switch between them with **[Ctrl]** **[↓]** **[↑]** **[→]** **[←]**.

For details on HUD and MFD modes, see sections 13 and 14.

## 12.2 External views

External views allow to have a look at any objects currently populating the simulated solar system, including the Sun, planets and moons, spacecraft, orbital stations and surface bases.

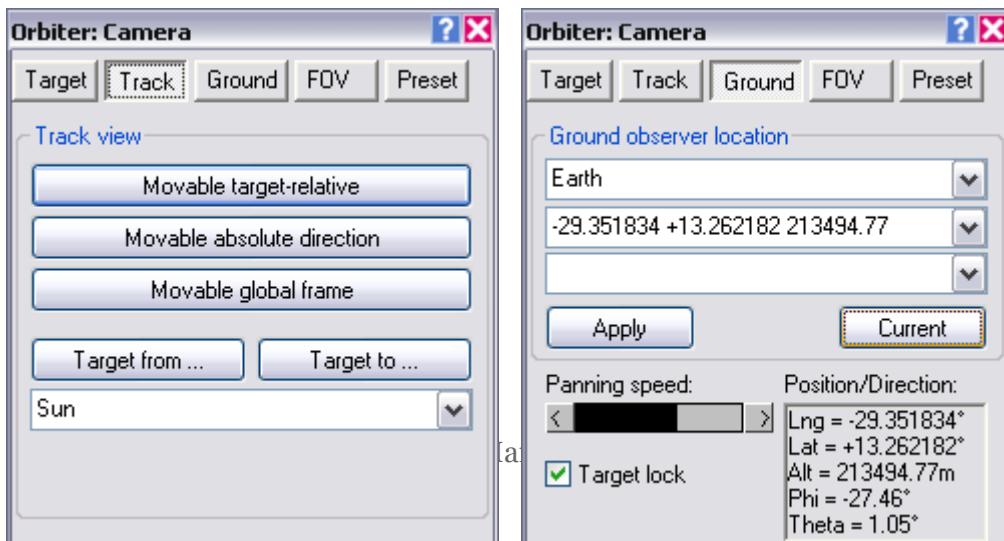
From cockpit view, an external view of the current spaceship can be selected by pressing **[F1]**. Other objects can be selected from the target list in the *Camera* dialog (**[Ctrl]** **[F1]**).

Two types of external camera modes are available:

**Track views** follow the object. The camera can be rotated around the target object by pressing **[Ctrl]** **[↓]** **[↑]** **[→]** **[←]** keys. The **[Page Up]** and **[Page Down]** keys move the camera towards or away from the target. Different camera panning modes for external views can be selected by pressing **[F2]** or via the *Track* tab in the Camera dialog:

- ▶ **Target-relative:** The camera is fixed in the target's local frame of rotation. Looking at a planet in this mode for example will rotate the camera together with the planet around its axis. **[Ctrl]** **[↓]** **[↑]** **[→]** **[←]** will rotate the camera around the target's local axes.
- ▶ **Global frame:** The camera is fixed in a non-rotating reference frame. Looking at a planet in this mode will show the planet rotating underneath the camera. **[Ctrl]** **[↓]** **[↑]** **[→]** **[←]** will rotate the camera around the axes of the ecliptic frame of reference.
- ▶ **Absolute direction:** This can be regarded as a mixture of the two modes above: The direction into which the camera points is fixed in an absolute frame, but it is tilted with respect to the target's local frame. **[Ctrl]** **[↓]** **[↑]** **[→]** **[←]** will rotate the camera around the target's local axes.
- ▶ **Target to ...:** Positions camera so that the specified object is behind the target.
- ▶ **Target from ...:** Positions camera so the specified object is behind the camera.

In *Target to ...* and *Target from ...* modes camera rotation (**[Ctrl]** **[↓]** **[↑]** **[→]** **[←]**) is deactivated, but radial camera movement with **[Page Up]** and **[Page Down]** is still available.



Camera: Selecting a track mode (left) or a ground-based view (right).

**Ground-based views** place the camera at a fixed point relative to the surface of a planet. This is a good way to follow the launch of a rocket from a spectator's perspective, or view the final approach of a Shuttle from the control tower. To select a ground-based view, select the *Ground* tab in the Camera dialog. You can now select one of the predefined observer locations from the lists, e.g. "Earth" + "KSC" + "Pad 39 Tower". Alternatively, you can just specify the planet and enter the location by hand, providing longitude (in degrees, positive towards east), latitude (in degrees, positive towards north), and altitude (in metres), e.g. "Earth" + "-80.62 +28.62 15". Click *Apply* to jump to the selected location.

You can also directly use the current camera location in ground observer mode, by clicking *Current*. The longitude, latitude and altitude are then entered automatically.

You can move the observer location by pressing  $\text{Ctrl} \downarrow \uparrow \rightarrow \leftarrow$ , and the observer altitude by pressing  $\text{Page Up}$  and  $\text{Page Down}$ . The speed at which the observer moves can be adjusted with the *Panning speed* slider in the dialog box, in the range from 0.1 to  $10^4$  m/s.

There are two ways to select the camera orientation: If the *Target lock* box in the dialog is ticked, the camera is always automatically pointing towards the current camera target. If the box is not ticked, the camera direction can be modified manually by pressing  $\downarrow \uparrow \rightarrow \leftarrow$ .

See also Section *Planets* in OrbiterConfig.pdf on how to add new observer sites to a planet definition file.

In external views a display of target parameters can be toggled by pressing  $\square$ .

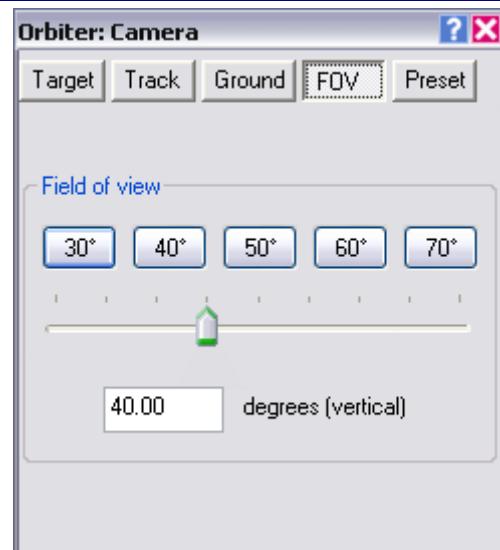
## 12.3 Selecting the field of view

The camera aperture defines the visible field of view (FOV). It can be adjusted in a similar way to the zoom function of a camera lens.

To set the aperture, select the *FOV* tab in the Camera dialog. The supported range is between  $10^\circ$  and  $90^\circ$  (Orbiter defines the field of view as the vertical aperture, between the top and bottom edge of the simulation window). The most natural aperture depends on the size of the simulation window on your screen, and the distance between your eyes and the screen. Typical values are between  $40^\circ$  and  $60^\circ$ .

You can adjust the field of view by clicking one of the aperture buttons, moving the slider, or entering a numerical value in the edit box.

The keyboard shortcuts are  $Z$  and  $X$  to continuously decrease or increase the FOV, respectively, or  $\text{Ctrl } Z$  and  $\text{Ctrl } X$  to decrease and increase the FOV in discrete steps



Camera: field of view selection

of  $10^\circ$ . The current field of view is displayed in the status section in the top left corner of the simulation window.

## 12.4 Storing and recalling camera modes

Orbiter provides an easy method to store and recall camera modes in a preset list. Click on the *Preset* tab in the Camera dialog. Any available modes are listed here. To activate a mode, double-click it in the list, or select the mode and click *Recall*.

To store the current camera mode as a new preset in the list, simply click *Add*. This will produce a new entry with a short description. To delete a mode, click *Delete*, or *Clear* to clear the whole list.

Each entry remembers its track mode, position, target and aperture. The preset list is a good way to prepare a set of camera angles beforehand (for example to follow a launch) and then activate them quickly without having to adjust the positions manually. The preset list is stored together with the simulation state, so it can be shared via a scenario file.



Camera: Mode preset list

## 13 Generic cockpit view

Generic cockpit mode displays flight information in a standard format and is available for all vessels. Some vessel types may additionally provide customised instrumentation in the form of 2-D panels or 3-D virtual cockpits. In that case, **F8** switches between the available modes.

The generic view mode represents a head-up display (HUD) that projects various avionics data displays directly onto the pilot's forward view.

The HUD is switched on/off with **Ctrl H**. HUD modes can be selected with **H**. The following modes are available:

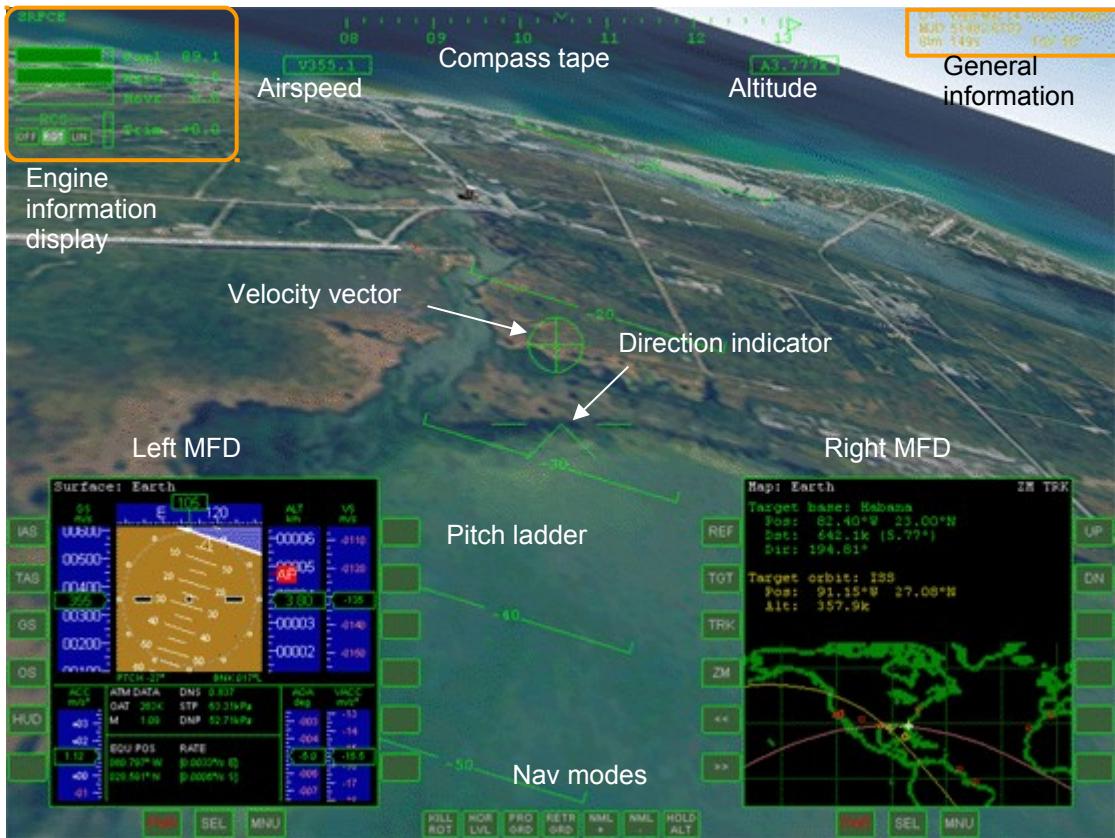
- ▶ **Surface:** Displays horizon pitch ladder, compass ribbon, altitude and “airspeed”.
- ▶ **Orbit:** Displays orbital plane pitch ladder, prograde and retrograde velocity markers
- ▶ **Docking:** Displays target distance and relative velocity markers.

All HUD modes show engine and fuel status in the top left corner, and general information (time and camera aperture) in the top right corner.

Two multifunctional displays (MFDs) can be displayed independent of the HUD mode (see Section 14). Each MFD has up to 12 mode-dependent function buttons along the left and right edges of the display, and 3 standard buttons below the display. The standard buttons are:

- ▶ **PWR:** Turns the MFD display on or off. This button is available even if the MFD is deactivated, provided that the HUD is activated.
- ▶ **SEL:** Displays the MFD mode selection screen. This allows to activate a different MFD mode. If more than 12 modes are available, use SEL repeatedly to show more modes.
- ▶ **MNU:** Displays an onscreen menu for the available function buttons of the current MFD mode, including the associated keyboard shortcuts.

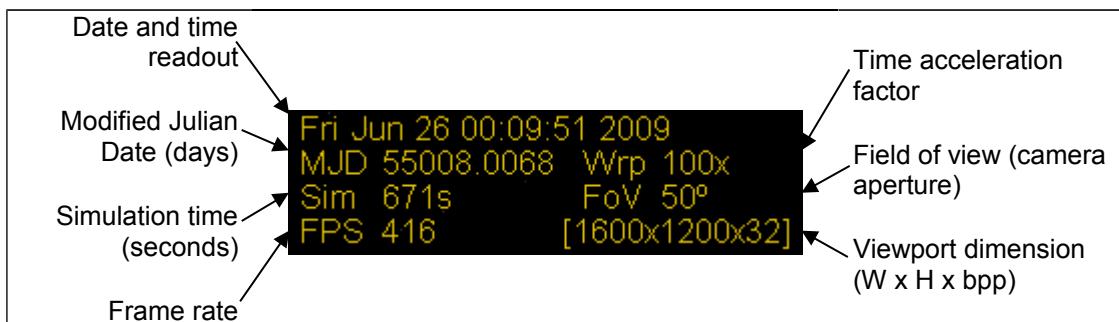
The MFD buttons can be operated either with the mouse, or with keyboard shortcuts.



*Generic cockpit view with two onscreen MFD displays and HUD in surface mode.*

### 13.1 General information display

A block of data with information about simulation time and speed, frame rate and camera aperture, displayed in the top right corner of the simulation window. This display can be turned on and off with .



*General simulation information*

The date in Orbiter is referenced to Barycentric Dynamical Time (TDB). TDB is a linear time scale (measured at the barycentre of the solar system for the purpose of accounting for relativistic effects) useful for expressing planetary motion and other celestial events. It is similar, but not identical, to Universal Time UT, which is the time reference that terrestrial clocks are generally referenced to (subject to a time zone offset). UT is adjusted to variations in Earth's rotation by inserting leap seconds at irregular intervals. Currently the offset between TDB and UT is 66.184 seconds.

**Date:** TDB date and time readout.

**MJD:** The *Julian Date* (JD) is the interval of time in days elapsed since 4713 BC January 1 at Greenwich mean noon. The *Modified Julian Date* (MJD) is the Julian Date minus 2 400 000.5. Since dates are referenced to TDB, a day is defined as consisting of 86400 seconds (SI), rather than as mean solar day.

**Sim:** Simulated time (in seconds) elapsed since the start of the simulation.

**Wrp:** Time acceleration factor. This field is not displayed for acceleration factor 1 (real time).

**FoV:** (vertical) field of view, i.e. viewport camera aperture.

**FPS:** current frame rate (frames per second)

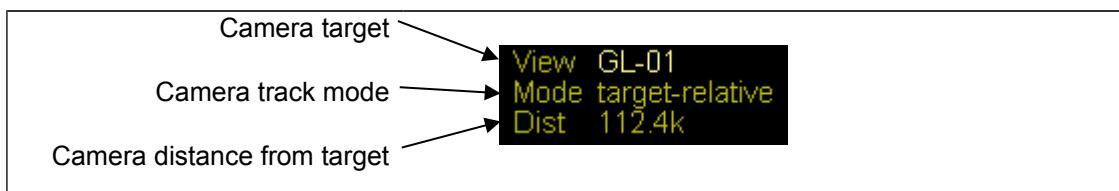
**Dim:** viewport dimension (width and height in pixels, colour depth in bits per pixel)

The display of frame rate and viewport dimension can be turned on and off with the **F** key.

Orbiter provides a date conversion utility (*date.exe*) in the *Utils* subdirectory. The *Scenario Editor* (see Section 20.1) allows to manipulate the date of a running simulation.

## 13.2 Camera target/mode display

This data block is displayed in external camera modes only, in the top left corner of the simulation window. It contains information about the camera target and track mode. This display can be turned on and off with **I**.



*External camera information*

**View:** Name of the current camera target.

**Mode:** The camera mode used for tracking the target.

**Dist:** Distance between camera and target.

## 13.3 Engine information display

The engine information display is only shown in non-panel cockpit views.

**Fuel status:** Remaining fuel is displayed as percentage of full tanks.

**Main engine:** The horizontal bar shows current main/retro engine thrust as fraction of max. engine thrust. Green indicates main thrusters, orange indicates retro thrusters. The numerical value shows acceleration in units of m/s<sup>2</sup> (positive for main, negative for retro thrust). Note that the acceleration may change even if the thrust setting doesn't, because the ship's mass changes as fuel is consumed.

**Hover engine:** If available, hover engines are mounted underneath the ship's fuselage to assist in surface flight, in particular during takeoff/landing. Display analogous to main engine.

**RCS indicators/controls:** The Reaction Control System (RCS) is an assembly of small thrusters arranged on the spacecraft so that they can be used for rotation and fine translational adjustments. The display shows the current mode (off/rotational/translational). The indicator buttons can be clicked with the mouse to change the RCS mode.

**Trim setting:** Displays the current setting of the trim control (if available). Trimming allows to adjust the flight characteristics during atmospheric flight.



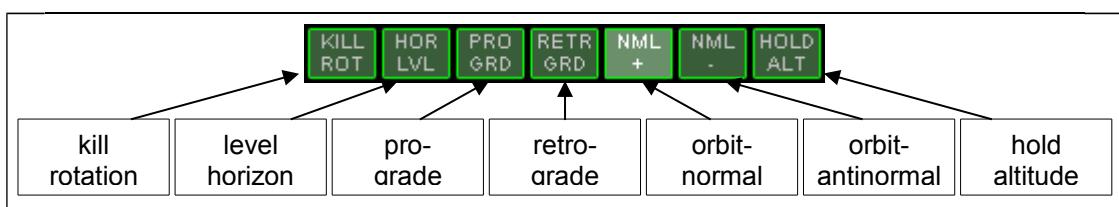
Fuel/engine displays and controls.

For more information about engines and spacecraft control see Section 15.

### 13.4 Navigation mode indicators/controls

The navigation mode indicators are shown as a row of buttons at the bottom edge of the generic cockpit view window. They display any active *navigation sequences* such as “prograde orientation” or “kill rotation”. The buttons can be clicked with the mouse to select or deselect modes. Note that some spacecraft types may not support all or any navigation modes.

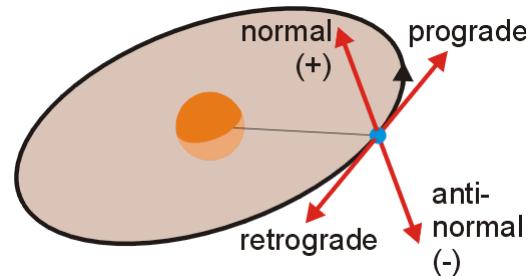
The navigation mode indicators are not shown if the HUD is deactivated.



mode	shortcut	action
KILLROT	[5] Num	kill any vessel rotation (auto-terminates)
HORLVL	[L]	keep vessel level with local horizon
PROGRD	[I]	align vessel with orbital velocity vector
RETRGRD	[J]	align vessel with negative orbital velocity vector
NML+	[;]	align vessel with normal of orbital plane
NML-	[']	align vessel with negative normal of orbital plane

HOLDALT	<input type="checkbox"/>	hold altitude (hover function)
---------	--------------------------	--------------------------------

All navigation mode except HOLDALT make use of the RCS. PROGRD, RETR-GRD, NML+ and NML- align the vessel into a specific attitude with respect to the orbital velocity vector and orbital plane, while HORLVL aligns with respect to the local horizon.



HOLDALT is only available for vessels that provide hover thrusters.

The KILLROT mode terminates automatically at zero angular velocity. All other modes are persistent and terminate only when deselected or when a conflicting mode is selected.

### 13.5 Surface HUD mode

Indicated by “SRFCE” in the upper left corner.

This mode displays a pitch ladder which indicates the ship’s orientation w.r.t. the current plane of the horizon. The plane of the horizon is defined by its normal vector, from the planet centre to the spacecraft.

The compass ribbon at the top of the screen indicates the ship’s forward direction w.r.t. geometric north. A marker shows the direction of the current target (spaceport).

The box left below the compass ribbon shows the current altitude [m]. The box right below the compass ribbon shows the current “airspeed” [m/s] (even if there is no atmosphere).

The surface-relative velocity vector direction is marked by “⊕”.

### 13.6 Orbit HUD mode

Indicated by “ORBIT Ref” in the upper left corner, where *Ref* is the name of the reference object.

This mode displays a pitch ladder relative to the current orbital plane, where the “o” line indicates the orbital plane. It also marks the direction of the orbital velocity vector (prograde direction) by “⊕” and retrograde direction by “+”. If neither the prograde nor retrograde direction is visible, then the direction of the ⊕ marker is indicated by a pointer labeled “PG” (prograde).

The reference object for the HUD can be manually selected by pressing   R.

### 13.7 Docking HUD mode

Indicated by “DOCK Tgt” in the upper left corner, where *Tgt* is the name of the target station.

This mode marks the current docking target (orbital station) with a square marker, and displays its name and distance. It also shows the direction and magnitude of the target-relative velocity vector. The velocity of the target relative to the ship is indicated by “ $\oplus$ ”. This is the direction in which you need to accelerate to synchronise your speed with the target. The opposite direction (the velocity of the ship relative to the target) is indicated by “ $+$ ”. If neither  $\oplus$  nor  $+$  are visible, then the direction of the  $\oplus$  marker is indicated by a pointer. Similarly, if the target marker is offscreen, its direction is indicated by a pointer.

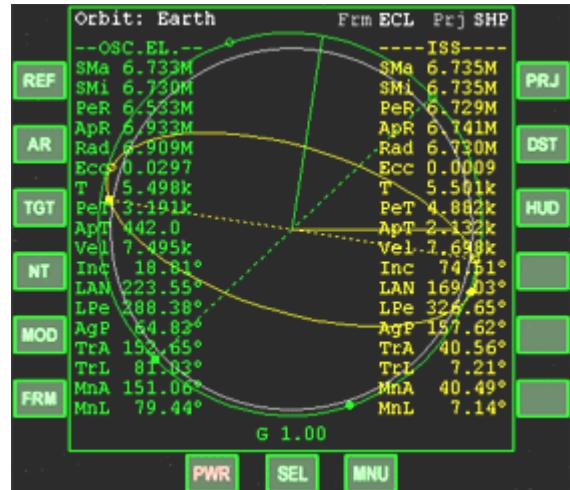
The target station for the HUD can be manually selected by pressing **Ctrl R**.

## 14 Multifunctional display modes

Multifunctional displays (or MFDs) are used in the cockpits of most military aircraft and modern airliners. They combine the function of a variety of traditional instruments in a compact format, and in combination with computerised avionics data processing present the pilot with situation-dependent relevant data.

In space flight, providing the pilot with information appropriate to the current flight regime is even more critical, and the Space Shuttle makes extensive use of MFD displays. Orbiter uses the MFD paradigm in a general and extendable way to provide flight data independent of vessel type.

An MFD is essentially a square computer display (e.g an LCD screen) and a set of input controls (usually push buttons arranged around the screen). The specific layout can vary, but the functionality is the same. The picture shows the MFD representation for the generic cockpit view mode which is available for all vessel types. Up to two MFDs can be displayed in this mode. Vessels which support customised 2-D instrument panels or 3-D virtual cockpits may use a different number of MFD screens. In generic mode, the displays are superimposed directly onto the 3-D scenery, representing for example a projection onto the pane of a HUD display in front of the pilot.



In the centre of the MFD is the data display. The 12 buttons along the left and right edge are mode-dependent function buttons. Their labels may change according to the current operation modus of the instrument. The three buttons along the bottom edge are static and mode-independent.

The MFDs can be operated either by left-clicking with the mouse on the buttons, or via the keyboard. All MFD keyboard functions are **Shift**-key combinations, where the left and right **Shift** keys operate the left and right MFD, respectively. For instrument panels with more than two MFD displays, only two can be operated with the keyboard; the others are limited to mouse control.

### Turning the MFD on and off

The **PWR** button activates and deactivates the MFD display. Keyboard shortcut is **Shift Esc**. In generic view mode, turning off the MFD also hides the buttons (except the power button, so it can be turned on again).

### Mode selection

The **SEL** button activates the mode selection screen. Keyboard shortcut is **Shift F1**. Each MFD mode provides information for a different navigation or avionics problem

(orbital parameters, surface parameters, docking and landing aids, etc.) For a full list of default modes see the following sections in this chapter. Many additional modes are available via 3<sup>rd</sup> party addons.

The display shows the available modes in the display area, one mode next to each function button. To select a mode, simply click the corresponding button. For selection with the keyboard, press the **Shift** key together with the mode selection key displayed in grey with each of the listed modes (for example, **Shift O** for Orbit mode).

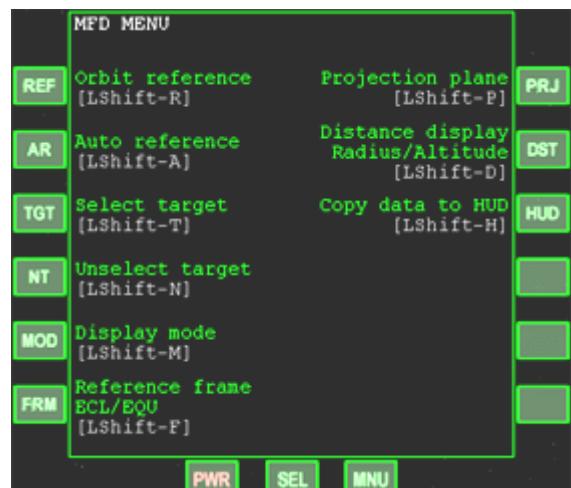
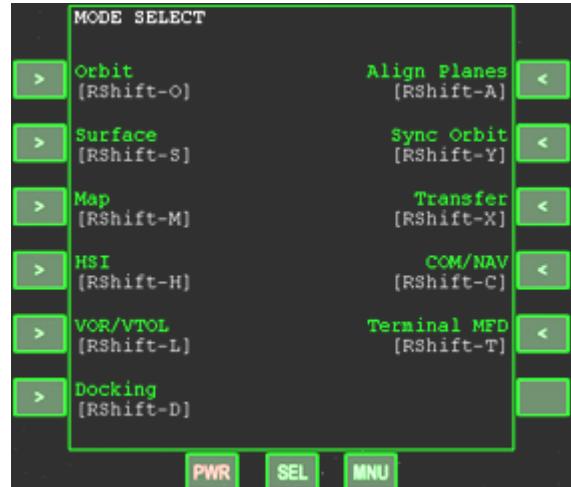
If there are more modes than can be displayed in a single page, pressing **SEL** (or **Shift F1**) repeatedly will page through all mode screens. Pressing **SEL** on the last mode screen will return to the previously selected MFD mode. Note that the mode selection with keyboard shortcuts works from any of the mode selection pages, even if the desired mode is not displayed on the current page.

### Function buttons

The function of the buttons to the left and right of the display depends on the current MFD mode, and their labels will change accordingly. Check the descriptions of the individual MFD modes in the following sections for the button functions of standard MFD modes. For addon modes, consult the accompanying documentation. In some cases the buttons may act as switches, where each press executes a specific function. In other cases it may be necessary to press down a key continuously to adjust a parameter.

Function buttons can also be activated with **Shift**-key combinations. Pressing the **MNU** button on the bottom edge of the screen will switch into menu mode (keyboard shortcut is **Shift [ ]**), where a short description of each function button is displayed, together with the associated keyboard key. Pressing **MNU** again (or pressing a function button) will restore the display.

In generic view mode, and in most instrument panels in Orbiter the MFDs have 12 function buttons, but in principle this could vary. If an MFD mode has defines more functions than can be assigned to the buttons, then pressing **MNU** repeatedly will page through the available sets of functions.



## Colour customization

The default colour schemes for MFD displays can be changed by editing the Config/MFD/default.cfg text file. Note that some addon MFD modes may override the default settings.

Below is a description of the standard MFD modes provided by Orbiter. See also the Quick MFD reference in 24.

### 14.1 COM/NAV receiver setup

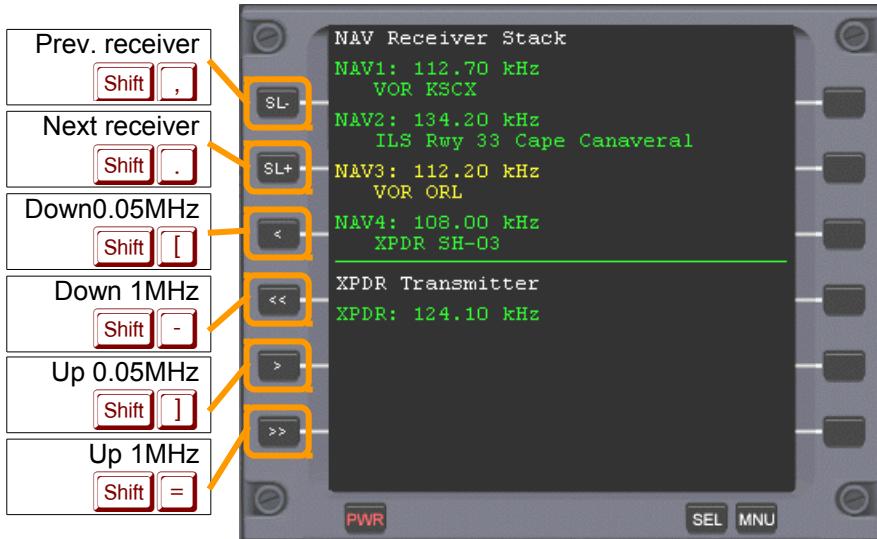
The *COM/NAV setup* MFD mode provides an interface to the ship's navigation radio receivers which feed data to the navigation instruments. It also allows to select the frequency of the ship's transponder which sends a signal to identify the vessel. The mode is activated via the *COM/NAV* entry from the MFD mode selection page ( 

The MFD lists frequency and signal source information for all NAV radios (NAV1 to NAVn). The number of receivers (n) depends on the vessel class. A NAV receiver can be selected from the list by   and  . The selected receiver is highlighted in yellow (see below).

#### Key options:

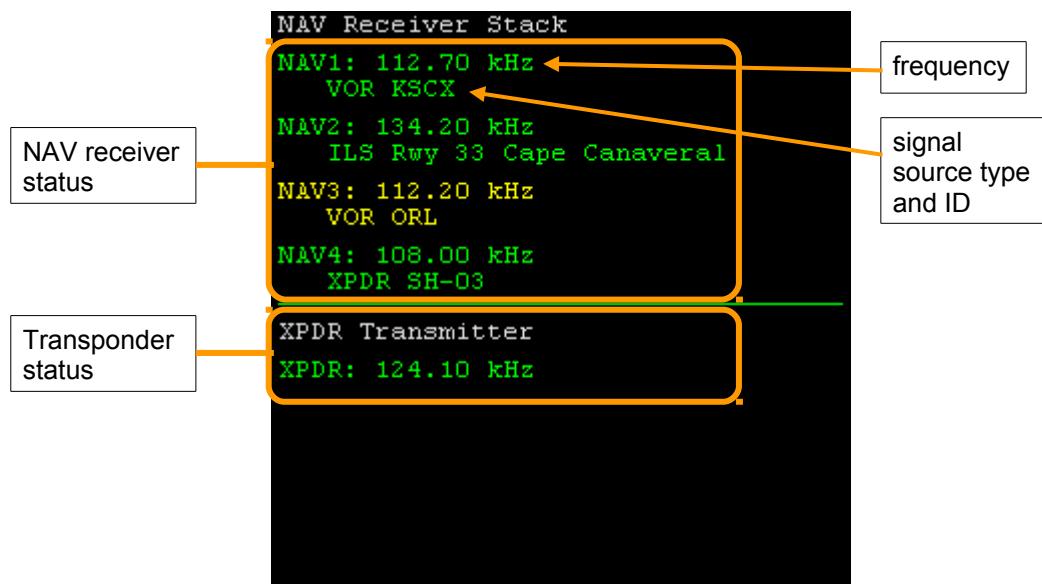
 	Select previous NAV receiver
 	Select next NAV receiver
 	Step down frequency 1 MHz.
 	Step up frequency 1 MHz
 	Step down frequency 0.05 MHz
 	Step up frequency 0.05 MHz
 	Scan frequency down
 	Scan frequency up

## MFD control layout:



## MFD display components:

The display is divided into two sections: The NAV receiver stack, listing the frequency and signal status of the ship's navigation radio receivers, and the Transponder status, showing the frequency of the ship's transponder.

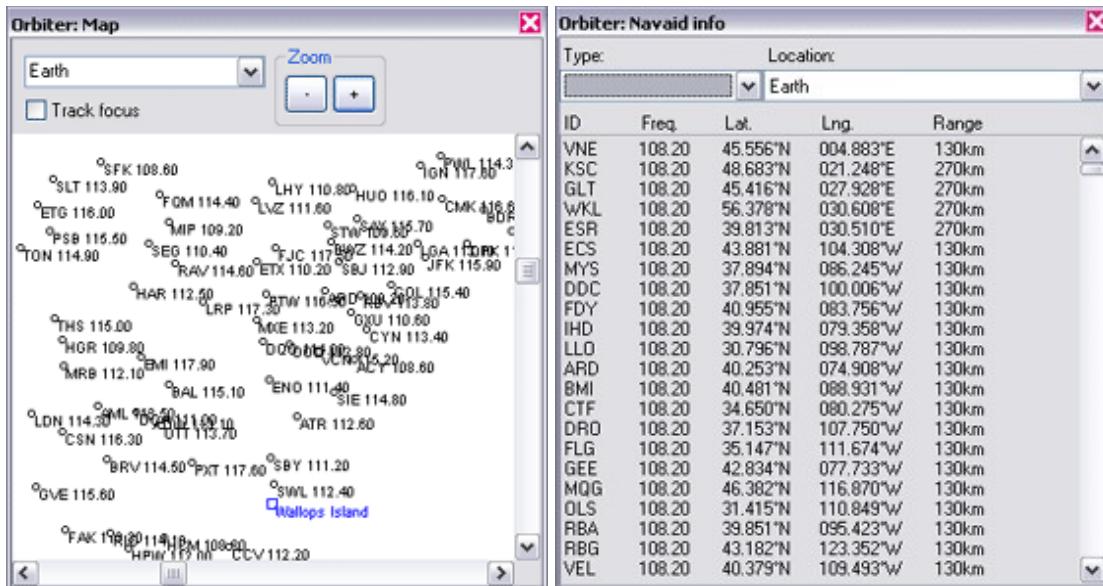


The frequency of the selected receiver/transmitter can be tuned in steps of 1MHz with **Shift** **-** and **Shift** **=**, and in steps of 0.05MHz with **Shift** **[** and **Shift** **]**, in the range from 85.00MHz to 140.00MHz. If a compatible NAV transmitter is within range, the instrument displays information about the signal source.

You can scan across the frequency range with **Shift** **Z** (down) and **Shift** **X** (up). Scanning will stop as soon as a signal is detected.

### Notes:

- ▶ Certain instruments such as the Launch/Land MFD mode are slaved to a NAV receiver, and will only work if a suitable signal is available. This behaviour differs from earlier Orbiter versions, where the data reference was obtained automatically.
- ▶ The *Object Info* (**Ctrl I**, see Section 11.2), *Navaid Info* (**Ctrl N**) and Map dialogs (**Ctrl M**) are a useful tools to obtain frequencies for navaid transmitters such as VOR and ILS beacons or vessel transponders.
- ▶ The positions and frequencies of VOR stations in your vicinity can also be displayed directly in the simulation window via the *VOR Markers* option of the *Visual helpers* dialog box (**Ctrl F9**).



Map and Navaid dialogs with VOR and ILS frequencies.

## 14.2 Orbit

The *Orbit* MFD mode displays a list of elements and parameters which characterise the ship's orbit around a central body, as well as a graphical representation. In addition, a target object (ship, orbital station or moon) orbiting the same central body can be selected, whose orbital track and parameters will then be displayed as well. The mode is activated via the *Orbit* entry from the MFD mode selection page (**Shift F1**).

The display shows the *osculating orbits* at the current epoch, i.e. the 2-body orbit corresponding to the vessel's current state vectors, with respect to a given celestial body. The orbital parameters may change with time due to the influence of perturbing effects (additional gravity sources, distortions of the gravitational field due to nonspherical planet shape, atmospheric drag, thruster action, etc.)

The orbital elements can be displayed with respect to one of two frames of reference: *ecliptic* or *equatorial*. The plane of the ecliptic is defined by the Earth's orbital plane, and is useful for interplanetary flights, because most planets orbit close to the ecliptic. The equatorial plane is defined by the equator of the current reference object, and is useful for low orbital and surface-to-orbit operations. Use **Shift F** to switch be-

tween the two frames of reference. The current mode is displayed in the top line of the display (Frm).

The plane into which the graphical orbit displays are projected can be selected via **Shift P**. The current projection plane is indicated in the top right corner of the instrument (Prj). *ECL* or *EQU* project into the plane of the ecliptic or equator, respectively. *SHP* projects into the vessel's current orbital plane, and *TGT* projects into the target's current orbital plane, if a target is specified.

The length of the current radius vector, and the apoapsis and periapsis distances can be displayed in two modes:

- ▶ planetocentric distance (distance from orbit focus), indicated by Rad, ApR, PeR, respectively
- ▶ altitude above planet mean radius, indicated by Alt, ApA, PeA, respectively.

Use **Shift D** to switch between the two modes.

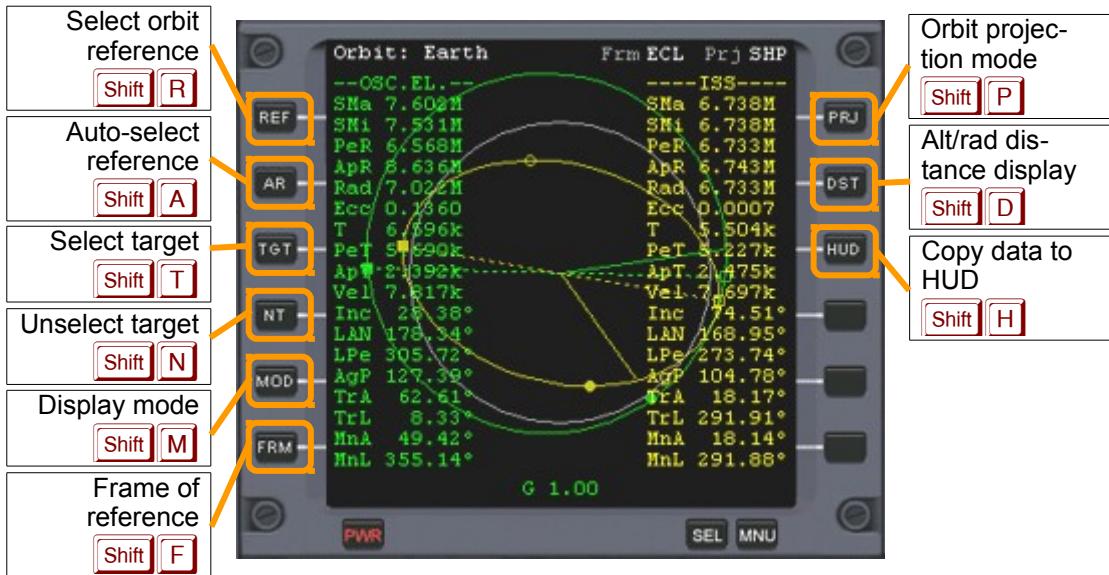
**Shift T** opens a menu to specify a target object. Only targets which orbit around the current reference object will be accepted. The target display can be turned off with **Shift N**.

Pressing **Shift H** will switch the vessel's head-up display to *Orbit* mode and copy the orbit reference object from the MFD to the HUD. This is often more convenient than selecting the HUD reference directly with **Ctrl R**.

### Key options:

<b>Shift A</b>	AR	Auto-select reference object.
<b>Shift D</b>	DST	Toggle radius, apoapsis and periapsis data display between planetocentric distance and altitude above mean planet radius.
<b>Shift F</b>	FRM	Toggle frame of reference (ecliptic, equator of reference object)
<b>Shift H</b>	HUD	Set HUD to Orbit mode and copy the current MFD orbit reference object.
<b>Shift M</b>	MOD	Toggle display mode (list only, graphics only and both)
<b>Shift N</b>	NT	No target orbit.
<b>Shift P</b>	PRJ	Toggle orbit projection mode (reference frame, ship's and target's orbital plane)
<b>Shift R</b>	REF	Select new reference object (planet or moon) for orbit calculation.
<b>Shift T</b>	TGT	Open menu for target selection.

## MFD control layout:

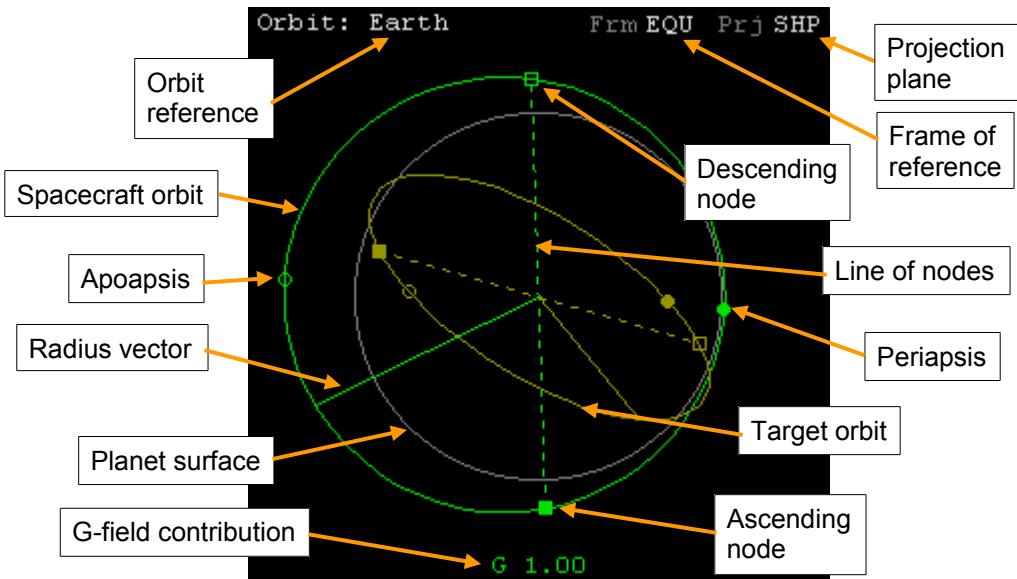


## MFD display components:

### 1. Graphic display mode

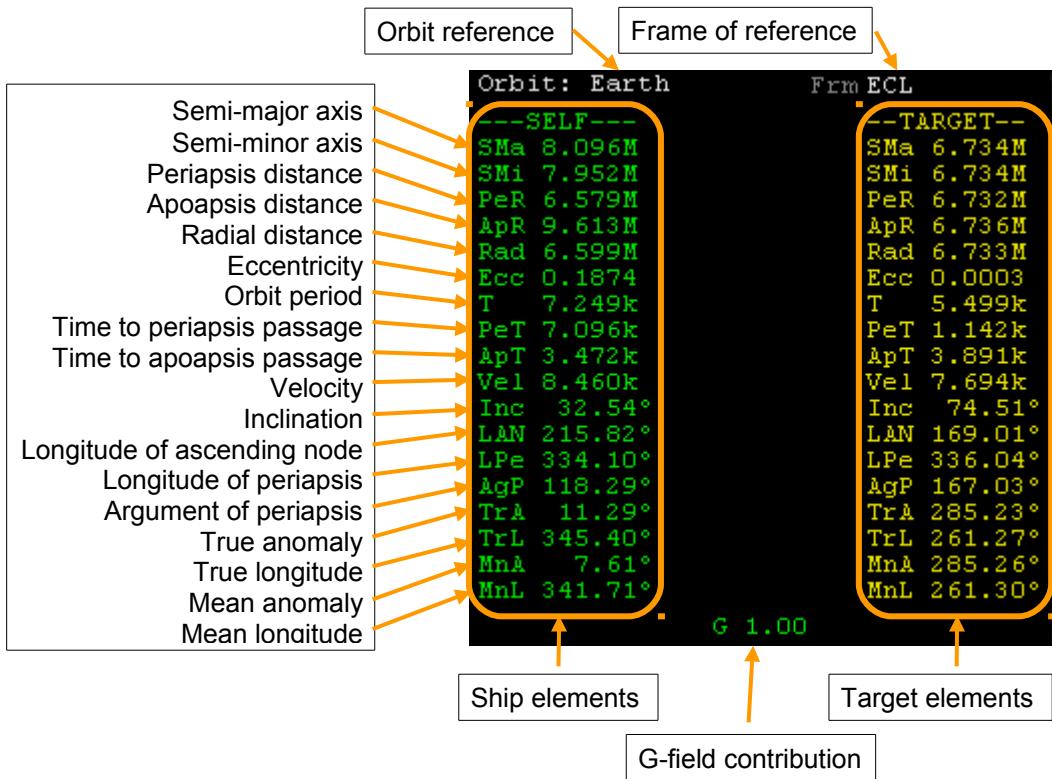
In graphical mode, the Orbit MFD shows the ship's orbit (green) and optionally the orbit of a target object (yellow) around the reference body (surface represented in gray). The display also shows the ship's current position (radius vector), the periapsis (lowest point of the orbit) and apoapsis (highest point), and the ascending and descending nodes w.r.t. the reference plane.

The user can select the plane into which the orbit representations are projected (orbital plane of the ship or target, ecliptic or equatorial plane).



## 2. Orbital element list mode

In list mode, the ship's orbital elements and other orbital parameters are listed in a column on the left of the MFD display (green). If a target is selected, its elements are listed in a column on the right of the MFD (yellow). The elements refer to the selected frame of reference, so they will change when switching between ecliptic (ECL) and equatorial (EQU) frame.



### Notation:

- ▶ Semi-major axis: the longest semi-diameter of the orbit ellipse.
- ▶ Semi-minor axis: the shortest semi-diameter of the orbit ellipse.
- ▶ Periapsis: The lowest point of the orbit (For Earth orbits, this is also called perigee. For solar orbits, it is also called perihelion).
- ▶ Apoapsis: The highest point of the orbit (for Earth orbits, this is also called apogee. For solar orbits, it is also called aphelion).
- ▶ Ascending node: The point at which the orbit passes through the reference plane (plane of the ecliptic, or equator plane) from below.
- ▶ Descending node: The point at which the orbit passes through the reference plane from above.
- ▶ Radius vector: The vector from the orbit's focal point to the current position of the orbiting body.

For further explanation of orbital elements see 26.

For hyperbolic (non-periodic) orbits, the following parameters are interpreted specially:

**SMa:** real semi-axis  $a$ : distance from coordinate origin (defined by intersection of hyperbola asymptotes) to periapsis. The semi-major axis is displayed negative in this case.

**SMi:** imaginary semi-axis  $b = a \sqrt{e^2 - 1}$

**ApD:** apoapsis distance: not applicable

**T:** orbital period: not applicable

**PeT:** time to periapsis passage; negative after periapsis passage

**ApT:** time to apoapsis passage: not applicable

**MnA:** mean anomaly, defined as  $e \sinh E - E$ , with  $E$  hyperbolic eccentric anomaly

### G-field contribution

The “G” value at the bottom of the display shows the relative contribution of the current reference body to the total gravity field at the ship’s position. This can be used to estimate the reliability of the Keplerian (2-body) orbit calculation. For values close to 1 a 2-body approximation is accurate. For low values the true orbit will deviate from the analytic calculation, resulting in a change of the orbital elements over time.

As a warning indicator, the G display will turn yellow for contributions < 0.8, and red if the selected reference object is not the dominant contributor to the gravity field. In that case, **[Shift] A** will select the dominant object.

## 14.3 VOR/VTOL

The VOR/VTOL MFD mode is a navigational instrument used for surface flight and vertical takeoff and landing. In addition to altitude and airspeed readouts it can display a graphical indicator of the relative position of a VOR (very high frequency omnidirectional range) navigation radio transmitter.

This MFD mode can be slaved to one of the ship’s NAV receivers. The current receiver and frequency is shown in the upper right corner of the display. If a signal is received, the transmitter ID is displayed in the second line. If the ship supports more than a single NAV receiver, a different receiver can be selected with **[Shift] N**. To set the receiver frequency, use the COM/NAV MFD mode (see section 14.1).

The instrument can also be used for vertical instrument landing (VTOL). When slaved to a VTOL transmitter, the target indicator shows the relative position of the corresponding launch pad.

### Key options:

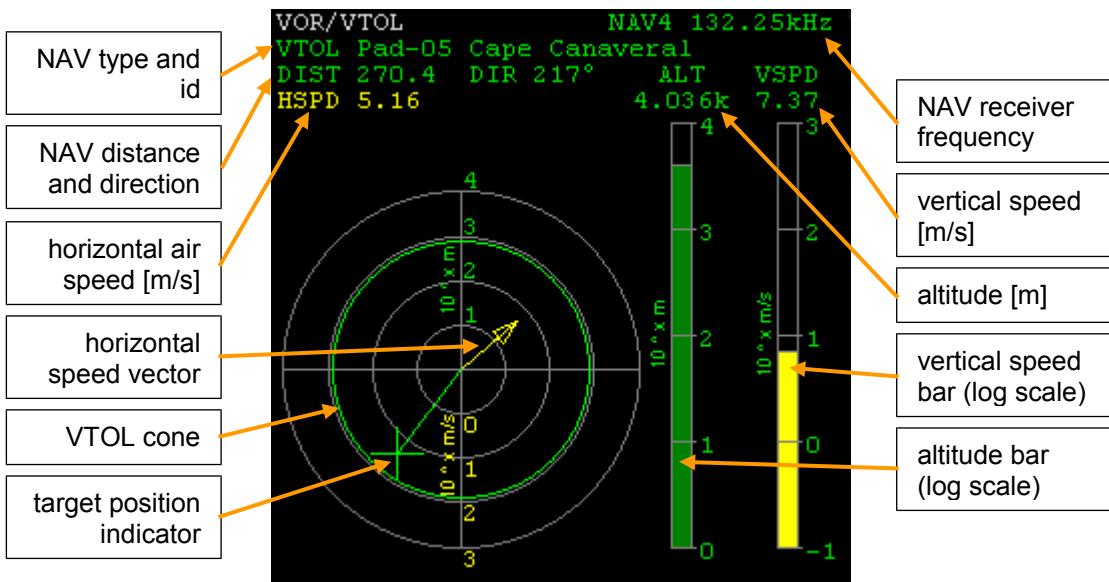
**[Shift] N**

Select navigation radio (NAV) receiver for VOR or VTOL information input.

## MFD control layout:



## MFD display components:



- ▶ **DIST:** distance to NAV transmitter [m]
- ▶ **DIR:** direction of NAV transmitter (ship-relative)
- ▶ **HSPD:** horizontal airspeed component [m/s]
- ▶ **ALT:** altitude [m]. The altitude bar has a range from 1 to  $10^4$ m (logarithmic scale).
- ▶ **VSPD:** vertical airspeed component [m/s]. The vertical speed bar has a range from  $\pm 0.1$  to  $\pm 10^3$ m/s (logarithmic scale). Positive vertical speed is indicated by a green bar, negative vertical speed by a yellow or red bar. *Red* is a surface impact warning.
- ▶ **Target indicator:** Shows the horizontal location of the slaved NAV transmitter (ship-relative) on a logarithmic scale. Range: 1 to  $10^4$ m.

- ▶ **Hspeed vector:** Shows the horizontal component of the airspeed vector (ship-relative) on a logarithmic scale. Range: 0.1 to  $10^3$ m/s.
- ▶ **VTOL cone:** This circle indicates the admissible deviation from the vertical touchdown vector as a function of altitude. During VTOL landing, the target indicator must remain inside the VTOL cone. A *red* circle indicates that the ship is outside the cone. The VTOL cone is displayed only when the MFD is slaved to a VTOL transmitter.

## 14.4 Horizontal Situation Indicator

The Horizontal Situation Indicator (HSI) consists of two independent displays. Each display can be slaved to a NAV receiver and show directional and relative bearing information. The instruments accept data from surface-based transmitters such as VOR and ILS. The function is similar to instrument navigation systems found in aircraft.

The display consists of a gyro-compass indicating the current heading at the 12 o'clock position. The yellow arrow in the centre of the instrument is the *course arrow* or *Omni Bearing Selector* (OBS). When the slaved NAV radio is tuned to a VOR transmitter, the OBS can be adjusted with the OB- ( ) and OB+ ( ) keys. For ILS transmitters, the OBS is automatically fixed to the approach direction.

The middle section of the course arrow is the *Course Deviation Indicator* (CDI). It can deflect to the left and right, to show the deviation of the OBS setting from the current bearing to the NAV sender. If the CDI is deflected to the left, then the selected radial is to the left of the current position.

In the lower left corner of the instrument is the TO/FROM indicator. "TO" means that you are working with a bearing from you *to* the ground station; "FROM" indicates a radial *from* the ground station to you.

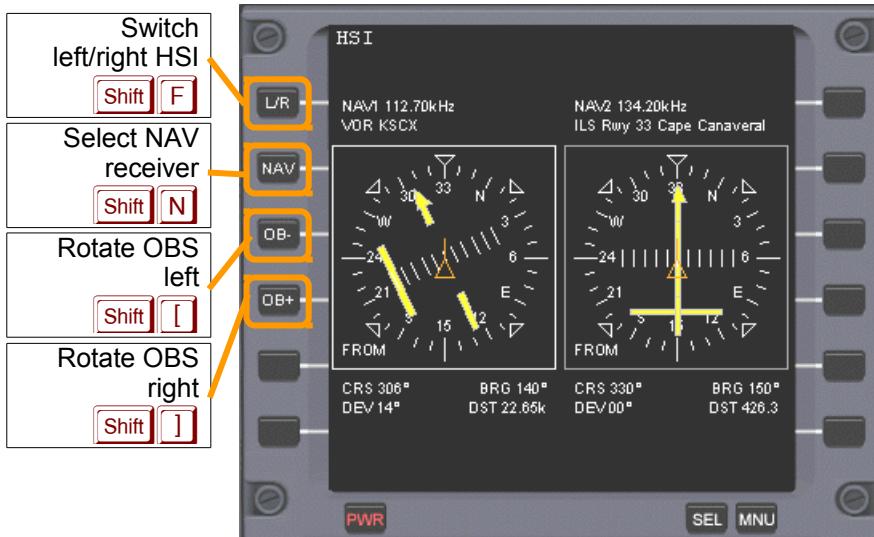
When tuned to an ILS (localiser) transmitter, the instrument shows an additional horizontal glideslope bar for vertical guidance to the runway. If the bar is centered in the instrument, you are on the correct glide slope. If it is in the upper half, the glide slope is *above* you, i.e. you are too low. If it is in the lower half, the glide slope is *below* you and you are approaching too high.

The refresh rate for the HSI MFD is 4Hz or the user selection in the Launchpad dialog, whichever is higher.

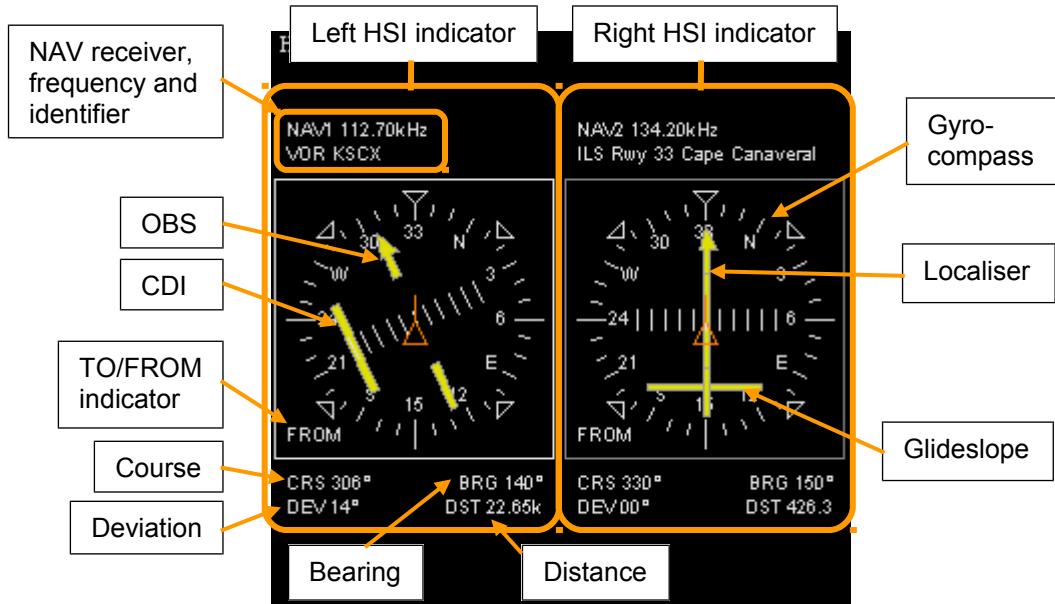
### Key options:

 	Select NAV receiver
 	Switch focus to left/right HSI instrument
 	Rotate OBS left
 	Rotate OBS right

## MFD control layout:



## MFD display components:



## To use the HSI for surface navigation:

- ▶ Determine the frequency of the VOR station you want to use (e.g. from the Map **Ctrl M** or spaceport info dialog **Ctrl I**) and tune one of your NAV receivers to that frequency (on the COM/NAV MFD)
- ▶ Slave one of the HSI displays to that receiver with **Shift N**.
- ▶ To fly directly towards the station, turn the OBS indicator until the CDI aligns with the arrow, and the TO/FROM indicator shows “TO”.
- ▶ Turn the spacecraft until the OBS indicator points to the 12 o'clock position.

- ▶ If the CDI wanders off to the left or right, turn the spacecraft in that direction until the arrow is aligned again.
- ▶ To fly away from the station, use the same procedure, but make sure that the TO/FROM indicator shows “FROM”.
- ▶ **To use the HSI for instrument landing:**
- ▶ Make sure the runway is equipped with ILS (use the spaceport info dialog, **[Ctrl] [I]**), and tune one of your NAV receivers to the appropriate frequency.
- ▶ Slave one of the HSI displays to that receiver.
- ▶ As soon as the ILS transmitter is in range, the OBS indicator will turn into the approach direction and can be used as a localiser indicator. At the same time, the glideslope indicator will become active. When both indicators are centered to form a crosshair, you are on course and on glideslope to the runway.

## 14.5 Docking

The *Docking* MFD assists during final approach to dock with another vessel or orbital station. It provides indicators for translational and rotational alignment with the approach path, as well as distance and closing speed readouts.

This instrument relies on docking approach data received by your spacecraft. Approach data can be acquired in three different modes:

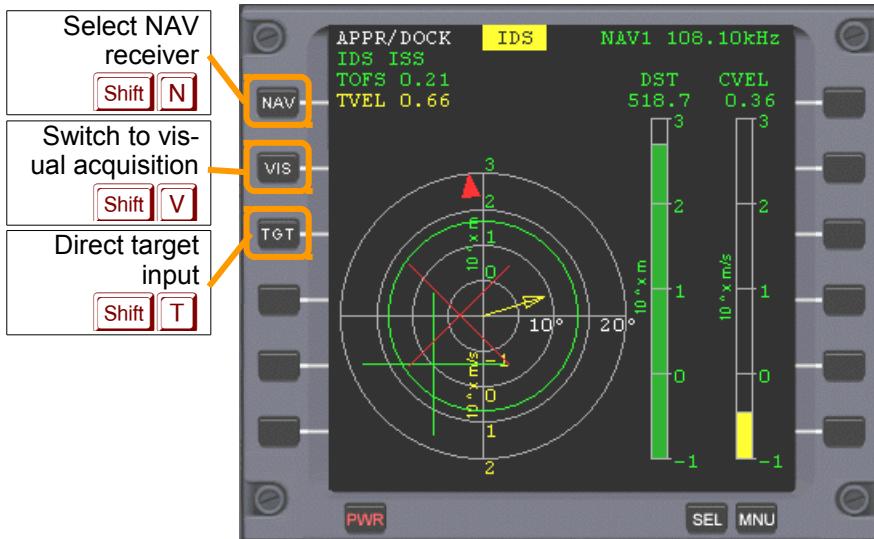
- ▶ IDS mode: data are acquired from a radio signal sent by the docking target. The IDS (Instrument Docking System) signal is obtained by tuning a NAV receiver to the appropriate frequency and slaving the Docking MFD to that receiver. The typical range for IDS is ~100km. To select a NAV receiver, press **[Shift] [N]**. The selected frequency is displayed in the upper right corner of the MFD.
- ▶ Visual mode: Docking parameters are acquired from onboard visual systems (typically video cameras mounted in the docking port). The visual system aids in docking to targets which don't provide IDS. The typical range for visual mode is ~100m. To switch to visual mode, press **[Shift] [V]**.
- ▶ Direct target selection: If you want to avoid the need to tune into a navigation transmitter signal, you can open target dialog (**[Shift] [T]**) and enter target name (and optional docking port index  $\geq 1$ ) directly. (This shortcut method may be dropped in a future version).

Apart from their different operational range, the three modes provide are identical in terms of the produced MFD display.

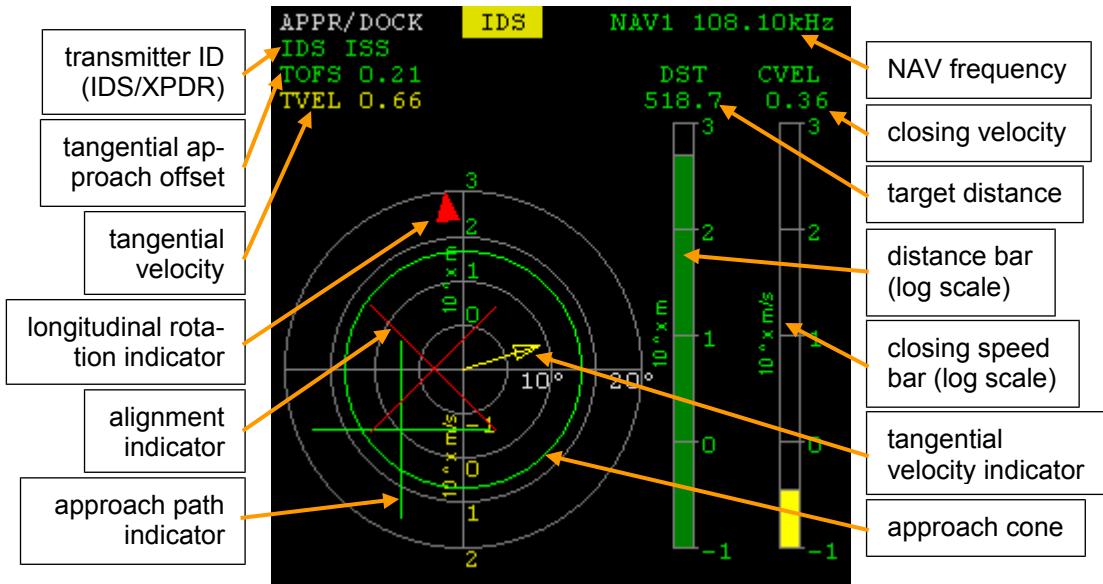
### Key options:

<b>[Shift] [N]</b>	Select NAV receiver for IDS information input.
<b>[Shift] [V]</b>	Switch to visual docking data acquisition mode.
<b>[Shift] [T]</b>	Direct target and docking port selection

## MFD control layout:



## MFD display components:



- ▶ **IDS source:** identifies the source of the currently received IDS signal.
- ▶ **TOFS:** tangential offset from approach path. This value is given in units of the approach cone radius at the current target distance. TOFS < 1 indicates a position inside the approach cone.
- ▶ **TVEL:** Tangential velocity (velocity relative to target, projected into plane normal to approach path) [m/s]
- ▶ **DST:** Dock-to-dock distance [m]. The bar shows the distance on a logarithmic scale in the range 0.1–10<sup>3</sup> m.
- ▶ **CVEL:** Closing speed [m/s]. The bar shows the closing speed on a logarithmic scale in the range 0.1–10<sup>3</sup> m/s. Yellow indicates positive closing speed.

The circular instrument shows the ship's alignment with respect to the approach path towards the allocated dock.

- ▶ **Approach path indicator:** The *green cross* indicates the position of the approach path relative to the ship. When centered, the ship is aligned on the approach path. The radial scale is logarithmic in the range  $0.1\text{--}10^3$  m. Tangential alignment should be performed with attitude thrusters in *linear* mode (see Section 15.2).
- ▶ **Tangential velocity indicator:** The *yellow arrow* indicates the relative tangential velocity of your vessel with respect to the target. The radial scale is logarithmic in the range  $0.01\text{--}10^2$  m/s. The numerical value is the tangential velocity [m/s]. *To align your ship with the approach path, engage linear attitude thrusters so that the arrow points towards the approach path indicator.*
- ▶ **Alignment indicator:** The *white/red cross* indicates the alignment of the ship's forward direction with the approach path direction. When centered, the ship's forward direction is parallel to the approach path. The cross turns red if misalignment is  $> 2.5^\circ$ . The radial scale is linear in the range  $0\text{--}20^\circ$ . Rotational alignment should be performed with attitude thrusters in *rotational* mode (see Section 15.2).
- ▶ **Longitudinal rotation indicator:** This arrow indicates the ship's longitudinal alignment with the docking port. To align, the indicator must be moved into 12 o'clock position by rotating the ship around its longitudinal axis, by engaging bank attitude thrusters in rotational mode (see Section 15.2). When alignment is achieved, the indicator turns white (misalignment  $< 2.5^\circ$ ). Note that this indicator is only displayed when directional alignment (see above) is within  $5^\circ$ .
- ▶ **Approach cone:** The concentric red or green circle indicates the size of the approach cone at the current dock distance. The ship should approach the dock so that the approach path indicator is always inside the approach cone (indicated by a green circle). The approach cone becomes smaller as the ship approaches the dock.

Closing speed should be reduced as the ship approaches the dock (using retro thrusters). The final speed should be  $< 0.1$  m/s.

#### Notes:

- ▶ To dock successfully, you must approach the dock to within 0.3 m. Additional restrictions may be implemented in the future (speed, alignment, etc.)
- ▶ No collision checks are currently performed. If you fail to dock and keep closing in, you may fly your ship through the target vessel.

## 14.6 Surface

The *Surface* MFD mode assists in flight close to planetary surfaces. It contains the following elements:

- ▶ **Artificial horizon** with pitch and bank readouts.

- ▶ **Heading indicator tape**
- ▶ **Altitude tape** with markers for perihel and aphel altitude
- ▶ **Vertical speed tape**
- ▶ **Vertical acceleration tape**
- ▶ **Speed tape** (IAS/TAS/GS/OS)
- ▶ **Acceleration tape**
- ▶ **Angle of attack tape**
- ▶ **Atmospheric data**
- ▶ **Equatorial position** (longitude and latitude, and rate of change)

The following atmospheric data are displayed (if applicable):

- ▶ **OAT**: Outside Air Temperature: Absolute atmospheric freestream temperature [K].
- ▶ **M**: Mach number  $M=v/a$ , with airspeed  $v$  and speed of sound  $a$ .
- ▶ **DNS**: Atmospheric density  $\rho$  [ $\text{kg m}^{-3}$ ]
- ▶ **STP**: Static pressure [Pa]
- ▶ **DNP**: Dynamic pressure  $q = \frac{1}{2} \rho v^2$  [Pa].

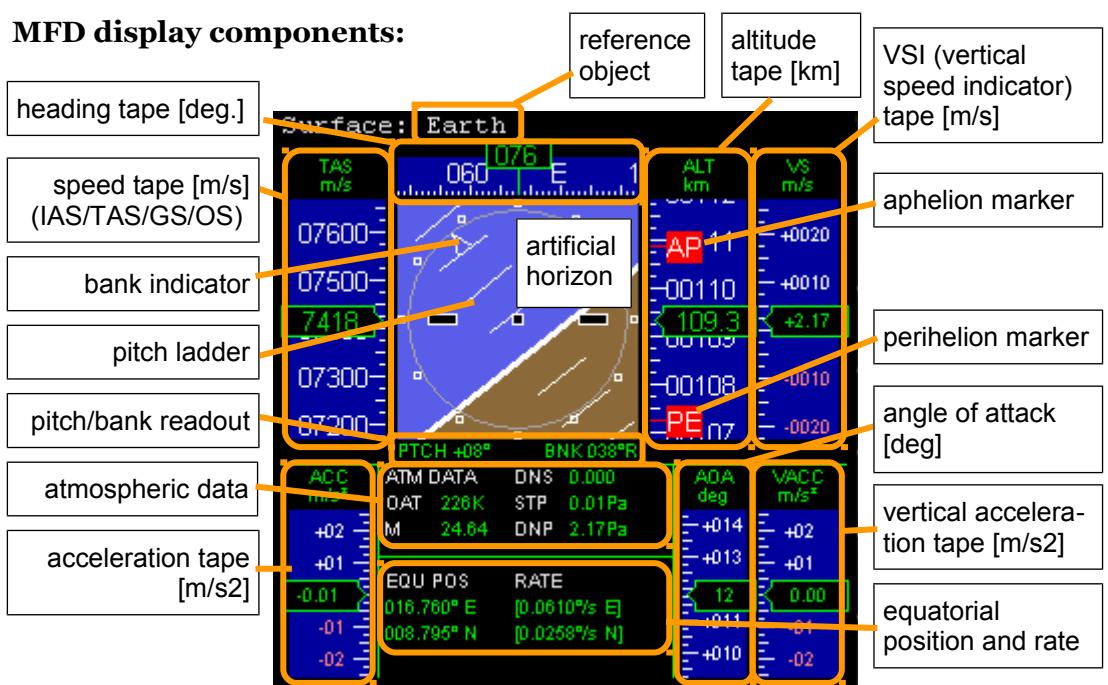
#### Key options:

	Select Indicated Airspeed display.
	Select True Airspeed display.
	Select Ground-relative Speed display.
	Select Orbital Speed display.

#### MFD control layout:



## MFD display components:



## Speed display modes:

The user can choose between four different speed indicator modes:

- ▶ **TAS (true airspeed):** The speed of the spacecraft relative to the surrounding atmosphere. Airspeed is usually measured with a pitot tube in the airstream recording the difference between freestream and stagnation point pressure. The TAS mode is only available if the freestream pressure  $p_1 > 10^{-4}$ Pa (on Earth, this corresponds to approx. 140 km altitude). If TAS cannot be measured, the speed tape is reset to 0 and the readout shows “----”.
- ▶ **IAS (indicated airspeed):** Commonly used in conventional aircraft. IAS is calibrated to atmospheric density and speed of sound at sea level. IAS and TAS are similar at low altitude, but start to diverge at higher altitudes, with IAS < TAS. The limit  $p_1 > 10^{-4}$ Pa also applies for IAS availability.
- ▶ **GS (ground-relative speed):** The magnitude of the vessel's velocity vector transformed into the rotating planet reference frame. This is similar to TAS at lower altitudes, but diverges at higher altitudes. Usually, TAS is no longer available at altitudes where the differences would become significant. *Note:* For an object in geostationary orbit, GS is zero since it is stationary relative to the rotating planet frame.
- ▶ **OS (orbital speed):** The vessel's velocity relative to the planet's centre in a non-rotating frame. This is identical to the “Vel” readout in the Orbit MFD. *Note:* OS is usually nonzero for a vessel at rest on the planet surface, since the planet itself rotates.

The speed tape left of the artificial horizon displays the vessel speed in the selected mode. The acceleration tape below shows the speed rate of change in the same mode.

The vertical speed and vertical acceleration tapes are not affected by the speed display mode.

The refresh rate for the Surface MFD is 4Hz or the user selection in the Launchpad dialog, whichever is higher.

**Technical background:** Orbiter uses a compressible flow model to calculate indicated airspeed:

$$v_{IAS} = a_s \sqrt{\frac{2}{\gamma - 1} \left[ \left( \frac{p_0 - p_1}{p_s} + 1 \right)^{\frac{\gamma-1}{\gamma}} - 1 \right]}$$

where  $p_0$  and  $p_1$  are the stagnation and freestream pressures, respectively,  $p_s$  and  $a_s$  are the standard sea level values for static pressure and speed of sound, and  $\gamma$  is the ratio of specific heats.

The stagnation point pressure  $p_0$  is obtained from the true airspeed by

$$v_{TAS} = a_1 \sqrt{\frac{2}{\gamma - 1} \left[ \left( \frac{p_0}{p_1} \right)^{\frac{\gamma-1}{\gamma}} - 1 \right]}$$

where  $a_1$  is the freestream speed of sound.

## 14.7 Map

The *Map* MFD mode shows a surface map of a planet or moon in a cylindrical (latitude vs. longitude) projection, and a superimposed orbit track of the spacecraft and an optional target object.

The Map MFD has been significantly improved and now supports the following features:

- ▶ **Choice of ground track or orbital plane display.** In ground track mode, the map shows the past surface track up to the current position, as well as a prediction of the future track line. In orbital plane mode, the map shows the great circle defining the intersection of the planet surface with the orbital plane.
- ▶ **Horizon lines.** The planet horizon, as seen from the spacecraft, can be displayed as a line. This defines the surface area currently visible from the spacecraft (or equivalently, the area within which the spacecraft appears above the horizon for a ground-based observer).
- ▶ **Track mode.** The map can either be scrolled manually, or set to track mode, where the spacecraft is kept in the centre of the display.
- ▶ **Terminator line.** The lit hemisphere of the planet can be marked by a shaded area or boundary line.
- ▶ **Vector coast and contour lines.** If provided for the target planet, the map can display coast lines or other contours, such as topological levels.
- ▶ **Large zoom range.** Zoom factors between 1 and 128 are supported.
- ▶ **Optional display of surface bases and navigation radio transmitters.** Surface bases and VOR transmitters can be displayed in the map. At higher zoom levels, the positions are labeled with names and frequencies.

- ▶ **Optional display of additional surface features.** If the target planet supports additional surface markers (see also Section 22.1), such as cities or geological features, these can be selected and displayed in the map.
- ▶ **Configuration page.** The Map MFD can be configured via a configuration page.

The current spacecraft position is displayed with a green cross. The ground track or orbit plane, as well as the visibility horizon, are shown as green lines. For ground track modes, the past track is shown in dark green, while the predicted future track is shown in bright green. In total, the track for approximately three orbits will be shown.

Note that in orbit plane display mode, the cross sectional line will slowly move across the map, as the planet rotates below it.

In addition to your own orbit, the position and ground track or orbit plane of a target object (e.g. a spacecraft or moon) orbiting the same central body can be displayed. The position and orbit lines are shown in yellow.

The positions of spacecraft and orbit target (longitude, latitude and altitude) are shown at the bottom of the map display.

Surface bases are indicated by yellow squares. A surface base can be selected as a target, which will display its position at the bottom of the page.

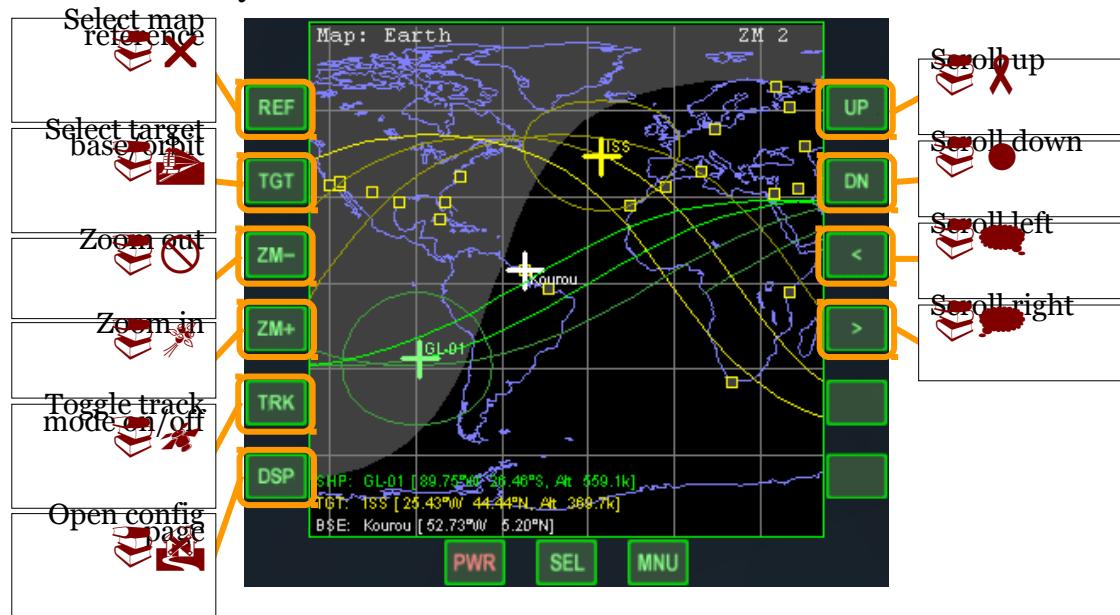
#### **Key options (map display):**

 R	REF	Open input box for reference planet/moon selection.
 T	TGT	Open a menu for target selection.
 X	ZM-	Decrease the zoom level by factor 2 down to 1x (global view).
 Z	ZM+	Increase the zoom level by factor 2 up to 128x.
 K	TRK	Switch automatic vessel track mode on or off.
 D	DSP	Switch to parameter selection page
 -	UP	Scroll map display up (not in track mode or global view).
 =	DN	Scroll map display down (not in track mode or global view).
 [	<	Scroll map display left (not in track mode).
 ]	>	Scroll map display right (not in track mode).

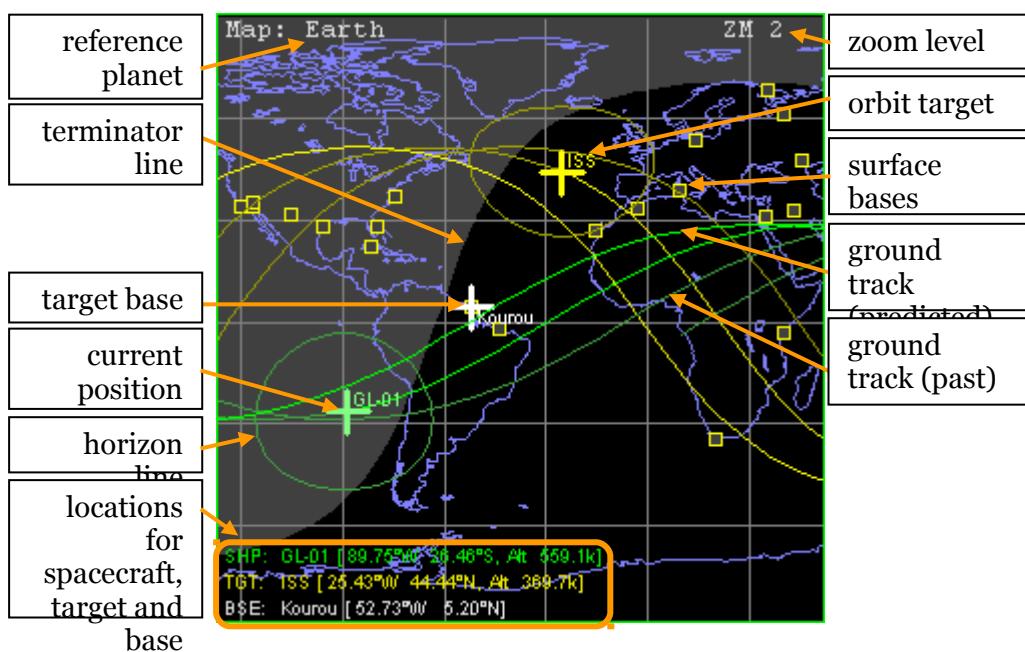
#### **Key options (parameter selection):**

 -	UP	Move selection marker up.
 =	DN	Move selection marker down.
 M	MOD	Modify the currently selected option.
 O	OK	Return to map display.

## MFD control layout:

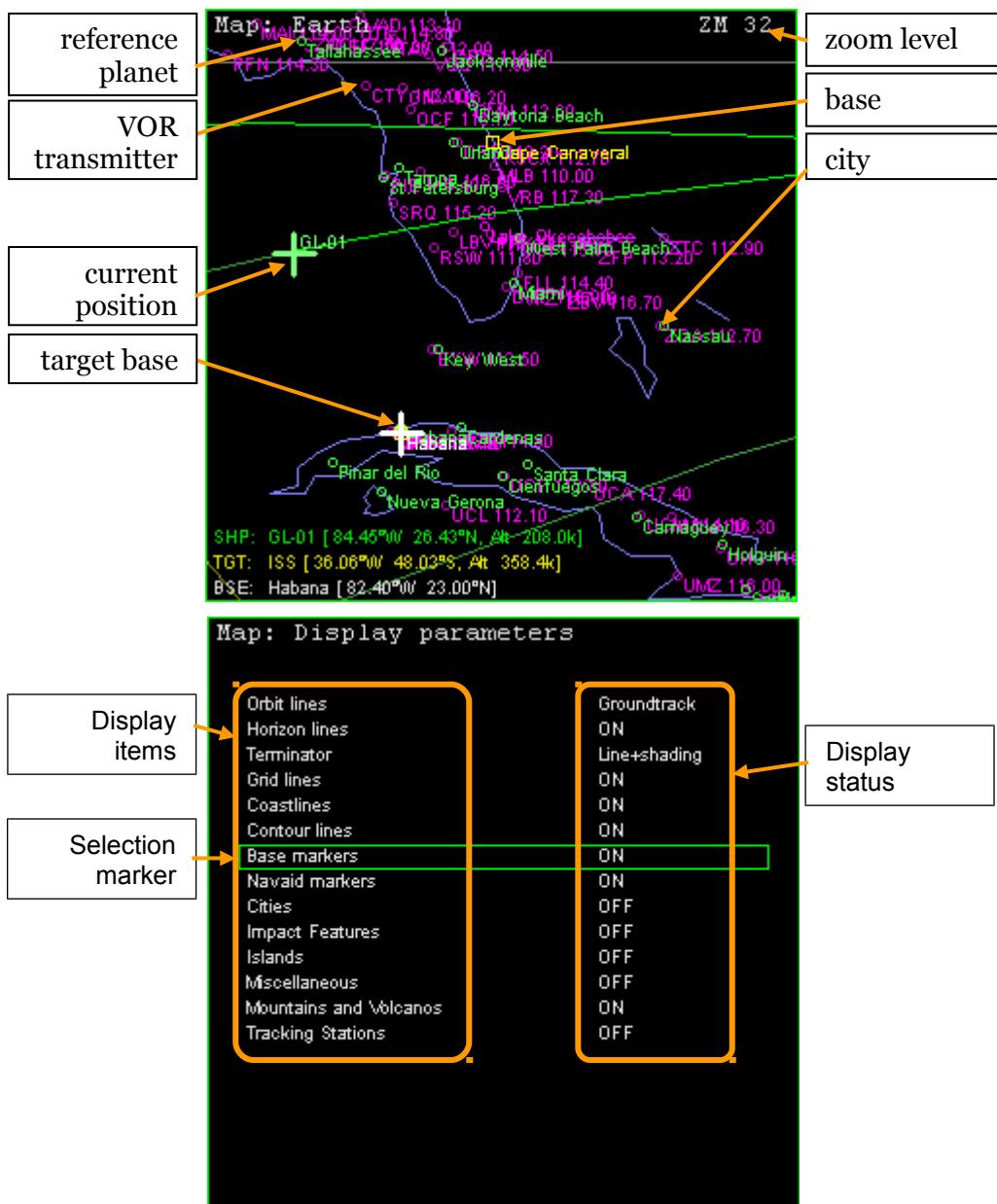


## MFD display components:



## Readouts:

- Spacecraft: position (longitude, latitude) and altitude
- Orbit target: position (longitude, latitude) and altitude
- Base target: position (longitude, latitude)



## Notes:

- ▶ Only objects (ships, stations or moons) orbiting the current reference planet will be accepted as orbit targets.
  - ▶ Only bases located on the current reference planet will be accepted as target bases.
  - ▶ Your ship's orbital plane will only be plotted if you are orbiting the current reference planet.
  - ▶ If required, the Map MFD mode can be reverted to the 2006 legacy version by adding the line "MFDMapVersion = 0" to the Orbiter.cfg configuration file in the main Orbiter folder.

## 14.8 Align orbital plane

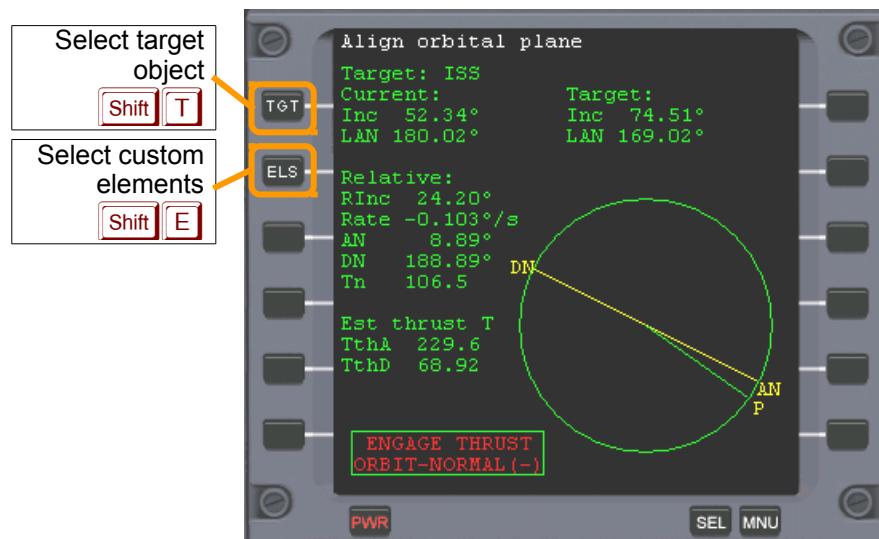
This MFD mode aids in rotating the orbital plane in space so that it corresponds with some target plane, e.g. the orbital plane of another object. The instrument contains the relevant orbital elements (inclination and longitude of the ascending node) of the current and target orbits. It also shows the relative inclination (angle between the two planes), the angles of the current radius vector towards ascending and descending nodes, the time to intercept the next node, and the predicted required thruster burn time. See section 17.4 on how to use this MFD mode.

The target plane can be either defined in terms of the orbital plane of another orbiting object, or by specifying the parameters that define the orientation of an orbital plane: the inclination and longitude of ascending node with respect to the ecliptic frame of reference.

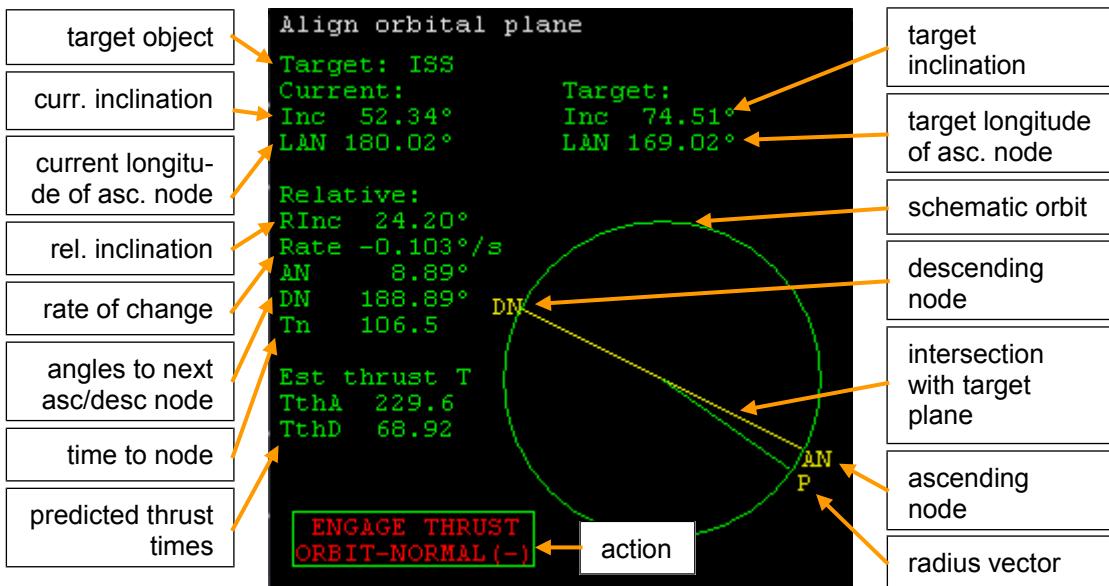
### Key options:

 T	Input a new target object or target orbital parameters.
 E	Input target plane as ecliptic inclination and longitude of ascending node.

### MFD control layout:



## MFD display components:



The MFD display shows a schematic orbit, indicating the directions of the ascending (AN) and descending (DN) nodes of the intersection of the current orbit with the target orbit, as well as our current position (P) along the orbit. The angular distances from the current position to the next AN and DN passages are shown on the left (range 0-360°). Also shown is the time to the next node passage (Tn).

Readouts for the relative inclination between the current and target orbits (*RInc*) and the rate of change of the relative inclination,  $dRInc/dt$  (*Rate*) help with timing the alignment burn.

Finally, the estimated burn times required to align the orbit with the target plane are listed, assuming a main engine burn at full thrust, perpendicular to the orbital plane.

Note that the required velocity change (Delta-V), and thus the burn time, depends on the orbital velocity, and may therefore be different at the ascending and descending nodes, if the orbit is not circular. The MFD shows the burn times both for the ascending (*TthA*) and descending nodes (*TthD*).

**Tip:** It is often more fuel-efficient to make the orbit more eccentric before applying the plane change, so that the radius distance of one of the nodes is increased and the corresponding Delta-V decreased. In particular if the plane change is to be combined with other changes to the orbit, a careful planning of the sequence of burns can help to minimise the fuel expenditure.

## 14.9 Synchronise orbit

The Synchronise Orbit MFD assists in catching up with an orbiting body once the orbital planes have been aligned (see previous section).

The instrument displays the ship's and target body's orbits, together with a reference axis and lists the times it will take both objects to reach this axis for a series of orbits.



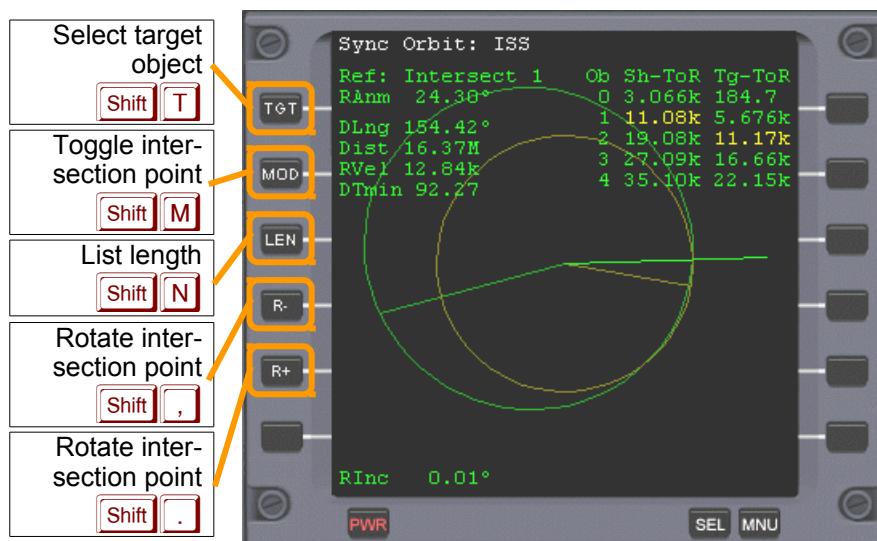
For this instrument to work properly the orbital planes of both objects must coin-

cide. The relative inclination of the orbital planes is shown in the lower left corner ("RInc"). If this becomes greater than  $1^\circ$ , realign the planes using the *Align Orbital Planes* MFD. Once the planes are aligned, all subsequent maneuvers should be performed in this plane.

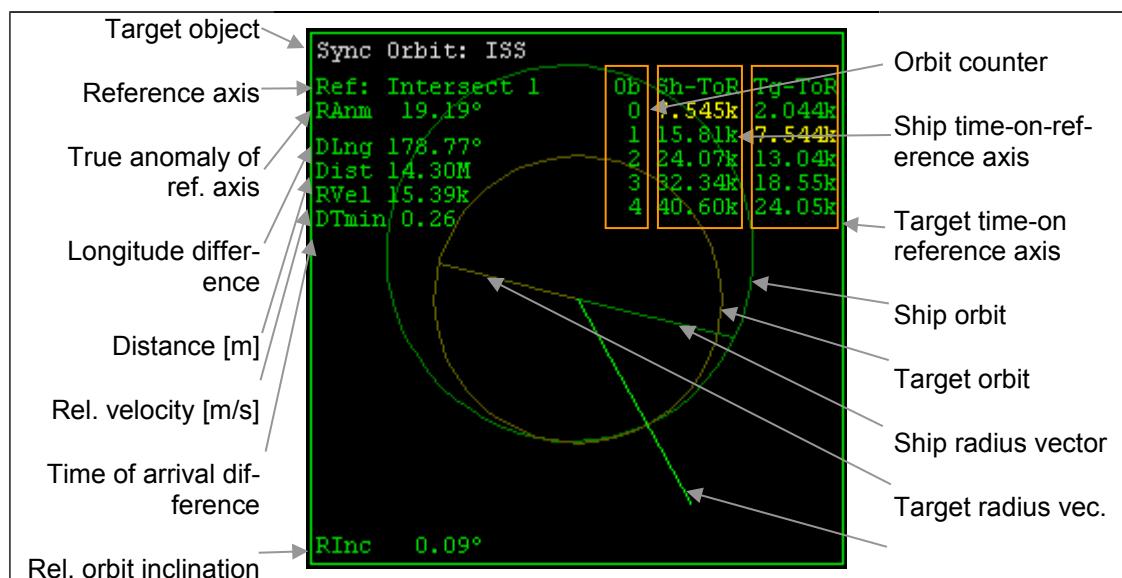
### Key options:

	Select target object. Only objects orbiting the same body as the ship will be accepted.
	Select reference axis mode. Intersection 1 and 2 are only available if the orbits intersect.
	Rotate reference axis (manual axis mode only).
	Select number of orbit timings in the list.

### MFD control layout:



### MFD display components:



*Synchronise Orbit MFD mode.*

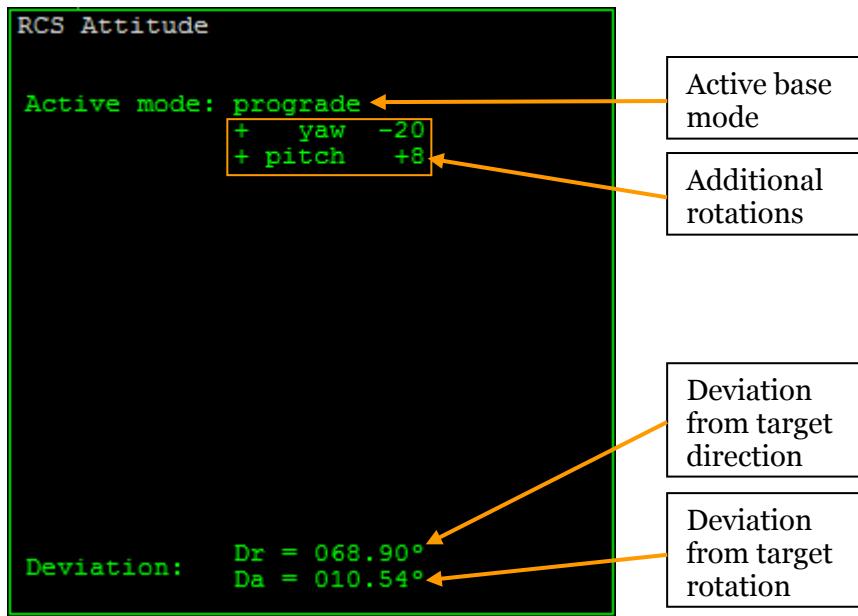
- ▶ **Target object:** The synchronisation target is displayed in the title line. It can be selected with **[Shift] [T]**.
- ▶ **Reference axis:** A selectable axis for which timings are computed. Can be selected with **[Shift] [M]** from one of the following: orbit intersection 1 and 2 (if applicable), ship and target apoapsis and periapsis, and manual. The manual axis can be rotated with **[Shift] [,]** and **[Shift] [.]**.
- ▶ **True anomaly of ref. axis (RAnm):** The direction of the reference axis w.r.t. the ship's periapsis direction.
- ▶ **Longitude difference (DLng):** The angle between ship and target as seen from the central body.
- ▶ **Distance (Dist):** Distance between ship and target [m].
- ▶ **Rel. velocity (RVel):** Relative velocity between ship and target [m/s].
- ▶ **Time-of-arrival difference (DTmin):** This is the minimum time difference [s] between the ship's and target's arrival at the reference point for any of the listed orbits (see below).
- ▶ **Rel. orbit inclination (RIInc):** Inclination between ship's and target's orbital planes.
- ▶ **Time-on-reference lists (Sh-ToR and Tg-ToR):** A list of time intervals for the ship and target to reach the selected reference point. The number of orbits can be selected with **[Shift] [N]**. The closest matched pair of timings is indicated in yellow. The DTmin value refers to this pair.

For usage of this MFD mode in orbit synchronisation, see Section 17.5.

## 14.10 RCS Attitude

The Attitude MFD mode provides advanced functions for orbital attitude control beyond the basic navigation modes described in Section 13.4. This MFD mode is an example for a script-driven MFD definition. In order to use it, the *ScriptMFD* module must be activated.

This is a relatively complex multi-page MFD mode. The main page shows the currently active attitude mode:



To create a new attitude mode, press the SET button. This opens the Mode definition page. If an attitude mode is currently active, the definition page initially displays the parameters of this mode. Otherwise, a default prograde mode is shown. You can use the BAS button to page through the available base modes: prograde, normal, perpendicular and radial. Each mode can be inverted by pressing the INV button. For example, this switches between prograde and retrograde, normal and anti-normal, etc. The following table defines the orientation of each base mode via two principal vessel axes (forward,  $+z$ , and up,  $+y$ ), relative to the orbital velocity vector ( $\mathbf{v}$ ), radius vector ( $\mathbf{r}$ ) and orbital plane normal ( $\mathbf{n} = \mathbf{r} \times \mathbf{v}$ ).

Mode	Vessel orientation
prograde	$+z \rightarrow +\mathbf{v}$ , $+y \rightarrow +\mathbf{n}$
retrograde	$+z \rightarrow -\mathbf{v}$ , $+y \rightarrow -\mathbf{n}$
normal	$+z \rightarrow +\mathbf{n}$ , $+y \rightarrow -\mathbf{v}$
antinormal	$+z \rightarrow -\mathbf{n}$ , $+y \rightarrow -\mathbf{v}$
perpendicular (in)	$+z \rightarrow -\mathbf{v} \times \mathbf{n}$ , $+y \rightarrow +\mathbf{v}$
perpendicular (out)	$+z \rightarrow +\mathbf{v} \times \mathbf{n}$ , $+y \rightarrow -\mathbf{v}$
radial (down)	$+z \rightarrow -\mathbf{r}$ , $+y \rightarrow +\mathbf{v}$
radial (up)	$+z \rightarrow +\mathbf{r}$ , $+y \rightarrow -\mathbf{v}$

After defining the base mode, you can add additional rotations to modify the vessel orientation. Press the  $+R$  button to add a rotation. You can then select a rotation axis (pitch, yaw, roll) by pressing the AX button. Set a rotation angle with the  $+V$  and  $-V$  buttons. You can add multiple rotations, but note that the order is significant: Rotations do not commute. You can select a rotation with the UP and DN buttons. Rotations can be deleted with the  $-R$  button.

When you are satisfied with your mode, press the GO button to activate it. You can continue editing the mode, and activate the modifications by pressing GO again.

To return to the main page, press RTN.

### **Rotational docking alignment**

The attitude MFD can also align the vessel with a target docking port. From the main page, press the DCK button. This opens the Dock alignment page. Docking alignment is performed with data from an IDS (instrument docking system) transmitter. Make sure that one of your NAV radios is tuned to the IDS transmitter of the target dock. Use the NAV button to select the appropriate radio. Docking alignment can only be activated once an IDS signal is received. Press ACT to activate the alignment mode. You can then return to the main page with RTN.

Docking alignment mode is cancelled if the IDS transmitter goes out of range, or if the slaved radio is re-tuned.

Docking alignment also works for off-axis docking ports.

### **Pre-multiplying an angular offset**

Sometimes it is useful to apply an angular offset to all attitude modes. For example, the Space Shuttle's OMS engines are tilted by 15° against the longitudinal axis. The resulting thrust vector therefore points to -15° pitch. For a prograde OMS burn, the Shuttle needs to pitch up by 15° against the orbital velocity vector. This constant offset due to engine arrangement can be taken into account by the Attitude MFD.

Open the Configuration page by pressing CFG from the main page. You can now add angular offsets by pressing the ADD button. You can then set the rotation axis and angle similar to the attitude mode setup. For example for the Shuttle, add a pitch rotation of +15°. When done, press RTN to return to the main page. The rotational offset will be added to all attitude modes, except for the dock alignment mode.

## **14.11 Transfer**

The Transfer MFD mode is used for calculating transfer orbits between planets or moons (or more generally, between any objects with significantly different orbits, for which the Sync orbit MFD is not sufficient).

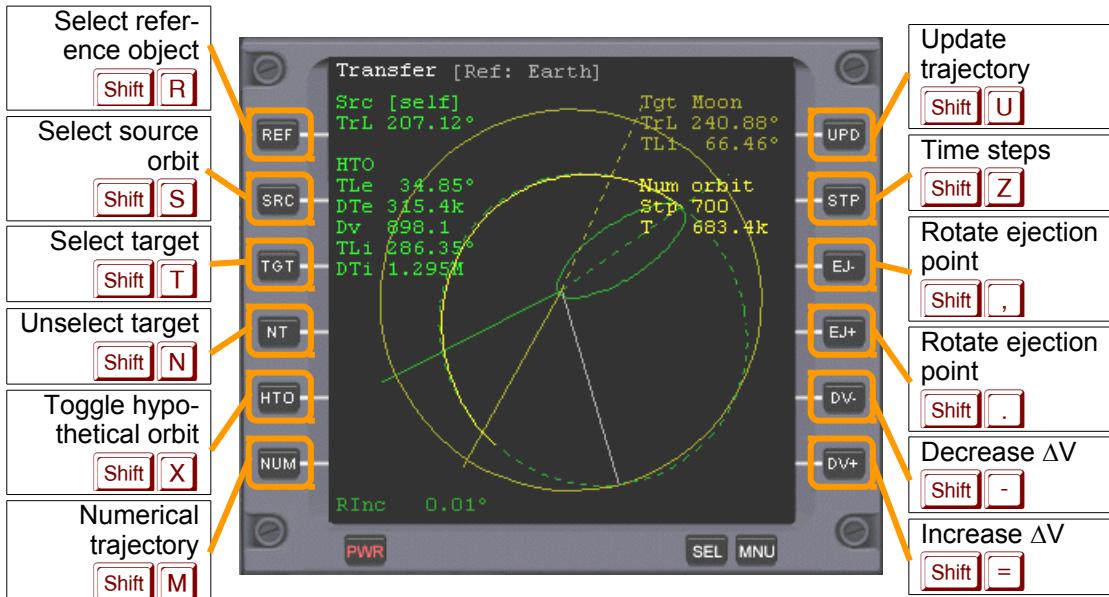
Note that Orbiter now contains Duncan Sharpe's *TransX* MFD mode as a plugin module, which supersedes and extends most of the Transfer MFD mode. *TransX* is described in a separate document (*TransXmanualv3*).

#### **Key options:**

 	Open input box for selection of reference celestial body.
 	Open a menu for source orbit object selection.
 	Open a menu for target selection.
 	Unselect target.
 	Toggle HTO (hypothetical transfer orbit) display on/off.
 	Toggle numerical multi-body trajectory calculation.
 	Refresh numerical trajectory, if displayed.

	Open input box for time step definition.
	Rotate transfer orbit ejection longitude.
	Decrease/increase ejection velocity difference.

### MFD control layout:



### MFD display components:

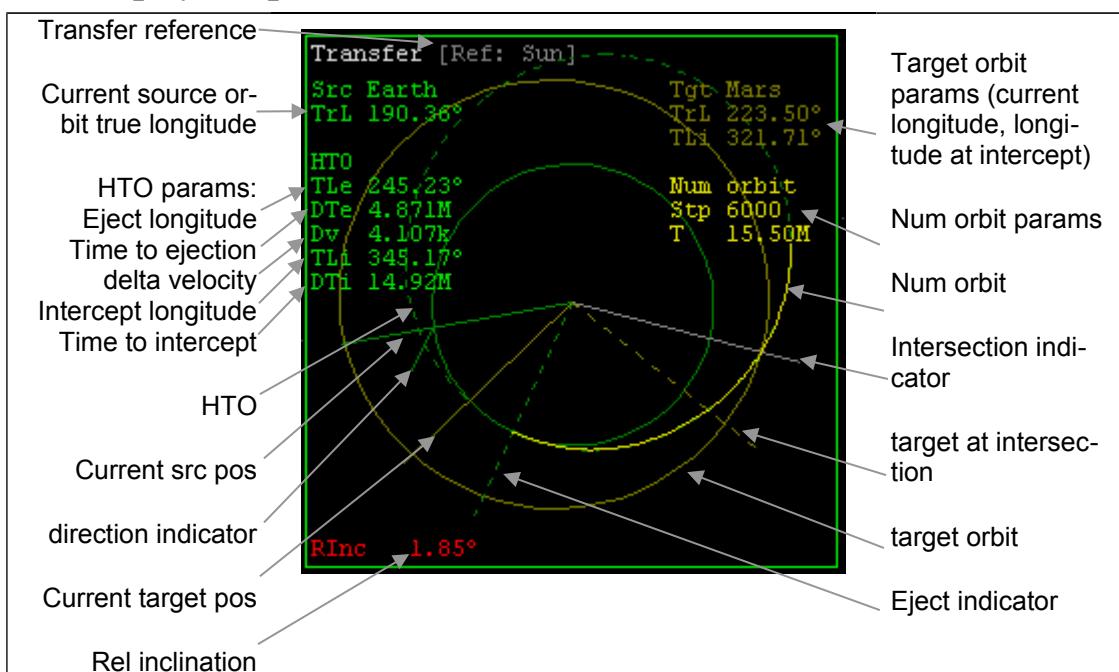


Figure 1: Transfer MFD mode.

The Transfer MFD looks similar to the Orbit MFD: it displays a *source* and a *target* orbit, relative to a selectable orbit *reference*. The source orbit is usually your ship's

current orbit, although sometimes a different source is more appropriate (see below). The MFD again assumes matching orbital planes of source and target, although this condition usually can not be precisely satisfied for interplanetary orbits.

### Source orbit selection

The source orbit is the orbit from which to eject into the transfer orbit. Usually the source orbit will be the ship's current orbit. In certain situations however it is better to use a different source. Consider for example an interplanetary transfer from Earth to Mars, using the Sun as reference. Since the ship's primary gravitational source will be Earth rather than the Sun, its orbit w.r.t. the Sun will be strongly distorted by the Earth's field. In this case it is better to directly use Earth as the source orbit.

Whenever the source is not identical to the ship, a small direction indicator will be displayed at the current source position which shows the ship's direction w.r.t. the source. This helps with timing the ejection burn (e.g. direction indicator pointing away from the Sun).

### Hypothetical transfer orbit

Unlike in Orbit mode, this MFD allows you to plot a hypothetical transfer orbit (HTO), which allows to set up "what if" scenarios, without having to change the actual orbit. The HTO display is toggled on/off via **Shift X**. It is calculated assuming that somewhere along the current source orbit a prograde or retrograde orbit ejection burn occurs. The HTO has two parameters: the *longitude* at which the ejection burn occurs (adjusted with **Shift , / .**) and the *velocity change* during the burn (adjusted with **Shift - / =**). The HTO is displayed as a dashed green curve in the MFD. The position of the ejection burn is indicated by a dashed green radius vector.

A number of parameters is shown when the HTO is turned on:

**TLe:** True longitude of orbit ejection point

**DTe:** Time to ejection point [s]

**Dv:** Velocity difference resulting from ejection burn [m/s]

**TLi:** True longitude of interception with target orbit (if applicable)

**DTi:** Time to interception with target orbit [s] (if applicable)

### Intercept indicator

If the source orbit (or, if shown, the HTO) intersects the target orbit, the intersection point is marked by a gray line, and the intersection longitude is displayed (TLi). The position of the target at the time when the ship reaches the intersection point is marked by a dashed yellow line. *The objective is to adjust the HTO so that the gray and dashed yellow lines coincide, so that ship and target arrive at the intersection point simultaneously.*

### Hohmann transfer orbit

A transfer orbit which just touches the target orbit (i.e. where ejection and intersection longitude are 180° apart) is called a Hohmann minimum energy transfer orbit,

because it minimises the amount of fuel used during the orbit ejection and injection points. Transfer orbits with larger major axis require more fuel, but are faster than Hohmann orbits.

### Ejection burn

Once the HTO has been set up, the ejection burn takes place when the ejection point is reached (when the solid and dashed green lines coincide). The ejection burn is pro-grade (or retrograde) given the orbit w.r.t. the current orbit reference. As the burn takes place, the current orbit (solid green line) will approach the HTO. The burn is terminated when the orbit coincides with the HTO, and Dv has reached zero. After ejection the HTO should be turned off so that intercept parameters are displayed for the actual transfer orbit.

### Numerical multi-body trajectory calculation

The source, target and transfer orbits discussed above are analytic 2-body solutions. The Transfer MFD however also supports a numerical trajectory calculation, to account for the effect of multiple gravitational sources. The display of the numerical trajectory is toggled with **Shift M**. The trajectory is displayed as a solid bright yellow line. The calculation is performed in discrete time steps, starting from the current source position, or (if displayed) from the HTO ejection point. Since the calculation of the trajectory can be time-consuming, it is not automatically updated, but can be refreshed with **Shift U**. The interval between time steps is automatically adjusted to provide consistent accuracy. The number of time steps, and thus the length of the trajectory, can be selected via **Shift Z**. The number of time steps, and the total time interval covered by the trajectory, are displayed under “Num orbit” in the MFD.

### Interplanetary transfers

Using the Transfer MFD for Earth to Moon orbits should be straightforward. For interplanetary transfers (e.g. Earth to Mars) a few caveats apply:

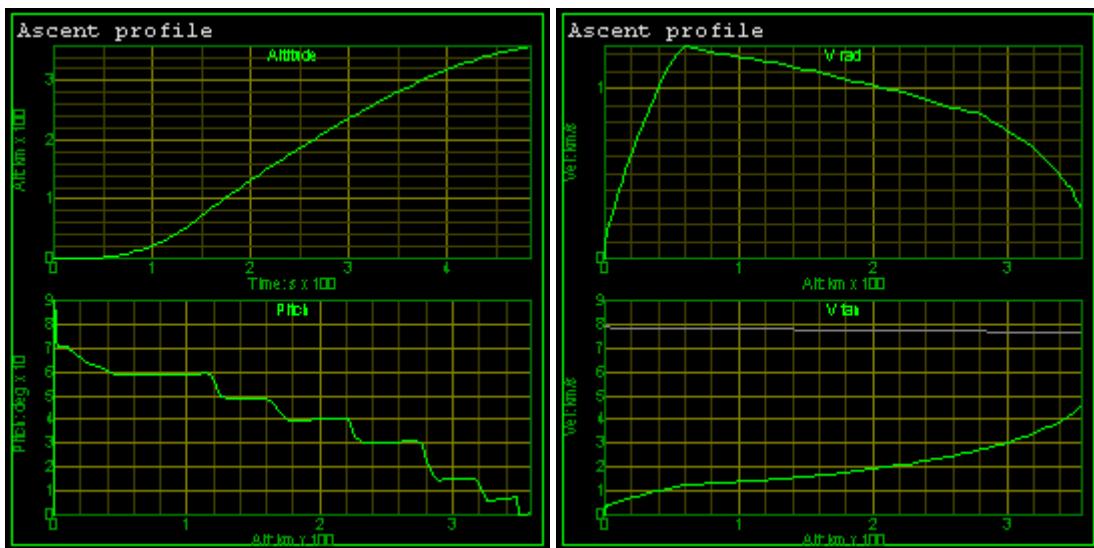
- ▶ For interplanetary transfers, the reference should be the Sun, and the source orbit should be the *planet currently being orbited*. This is because the ship’s orbit w.r.t. the Sun will be severely distorted by the planet.
- ▶ The ship should be in an orbit with zero inclination against the ecliptic before ejection. The relative inclination between source and target orbits cannot be adjusted, it is simply given by the relative inclination between the planets’ orbits.
- ▶ The ejection burn should take place with the Sun in opposition (on the planet’s “dark” side) so that the ship’s orbital velocity is added to the planetary velocity. This is the case when the source → ship direction indicator is pointing away from the Sun.
- ▶ Immediately before the ejection burn, switch the source orbit to your ship, so that Dv can be estimated.

## 14.12 Ascent profile (custom MFD mode)

This MFD mode is only available if the “Custom MFD” plugin is activated in the *Modules* section of the Launchpad dialog.

The ascent profile records a number of spacecraft parameters and displays them in graphs on the MFD. The following are recorded:

- ▶ Altitude as a function of time.
- ▶ Pitch angle as a function of altitude.
- ▶ Radial velocity as a function of altitude.
- ▶ Tangential velocity as a function of altitude.



Ascent profile MFD mode, pages 1 and 2

### Key options:

<input type="checkbox"/> Shift <input type="checkbox"/> P	Switch display page.
<input type="checkbox"/> Shift <input type="checkbox"/> A	Set altitude range.
<input type="checkbox"/> Shift <input type="checkbox"/> R	Set radial velocity range.
<input type="checkbox"/> Shift <input type="checkbox"/> T	Set tangential velocity range.

Parameters are sampled at 5-second intervals. A total of 200 samples are stored and cycled. By default, axis ranges are adjusted automatically, but manual range setting is possible.

### Circular orbit insertion

In the tangential velocity graph (Vtan), a gray line indicates the orbital velocity for a circular orbit as a function of altitude. If the vessel's tangential velocity crosses this line for a given altitude, while simultaneously the radial velocity crosses zero, circular orbit is achieved.

## 15 Spacecraft controls

This chapter contains guidelines on how to control your spacecraft in free space (outside the influence of aerodynamic forces due to an atmosphere). We are considering a “generic” vessel. Note that the handling of different spacecraft types may vary considerably. Always read the operating instructions of individual vessels, if available.

### 15.1 Main, retro and hover engines

*Main thrusters* accelerate the ship forward, *retro thrusters* accelerate it backward. Main and retro engines can be adjusted with  $\text{Ctrl} + \text{Num}$  (to increase main thrust or decrease retro thrust) and  $\text{Ctrl} - \text{Num}$  (to decrease main thrust or increase retro thrust). Main and retro thrusters can be killed with  $\text{Ctrl} * \text{Num}$ . The permanent setting can be temporarily overridden with  $+\text{Num}$  (set main thrusters to 100%) and  $-\text{Num}$  (set retro thrusters to 100%). If available, a joystick throttle control can be used to set main thrusters.

The ship’s acceleration  $\mathbf{a}$  resulting from engaging main or retro thrusters depends on the force  $\mathbf{F}$  produced by the engine and the ship’s mass  $m$ :

$$\mathbf{F} = m\mathbf{a}$$

Note that both  $\mathbf{a}$  and  $\mathbf{F}$  are vectors, that is, they have a direction as well as a magnitude. In the absence of additional forces (such as gravitation or atmospheric drag) the spacecraft will move with constant velocity  $\mathbf{v}$  as long as no engines are engaged. When engines are engaged, the ship’s velocity will change according to

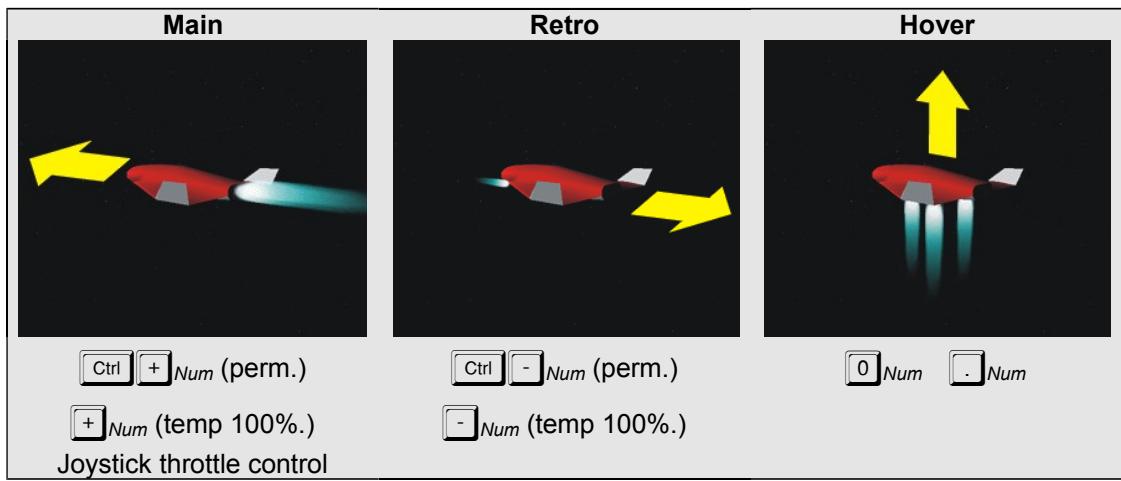
$$\frac{d\mathbf{v}(t)}{dt} = \mathbf{a}(t) \quad \text{or} \quad \mathbf{v}(t) = \mathbf{v}(t_0) + \int_{t_0}^t \mathbf{a}(t') dt'$$

Note that for a fixed thruster setting  $\mathbf{F}$  the acceleration will slowly increase as fuel is consumed, resulting in a reduction of the ship’s mass  $m$ .

Hover engines, if available, are mounted underneath the ship’s fuselage to provide upward thrust. Hover thrust is increased with  $0\text{Num}$  and decreased with  $.\text{Num}$ . Hover thrusters are useful to compensate for gravitational forces without the need to tilt the ship upward to obtain an upward acceleration component from the main thrusters.

The current main/retro thruster setting and corresponding acceleration is displayed in the upper left corner of the generic HUD (“Main”). The indicator bar is green for positive (main) thrust, and yellow for negative (retro) thrust. The hover thrust setting is also displayed if applicable (“Hovr”). The numerical acceleration value is in units of  $\text{m/s}^2$ . Spacecraft with customised instrument panels usually have their own indicators for thrust levels.

Spacecraft equipped with airfoils moving within a planetary atmosphere usually do not require hover thrusters except for launch and landing, because they produce an upward force (lift) when moving with sufficient airspeed, like a normal aircraft. Lift is speed-dependent and will collapse below a threshold speed (stall speed).



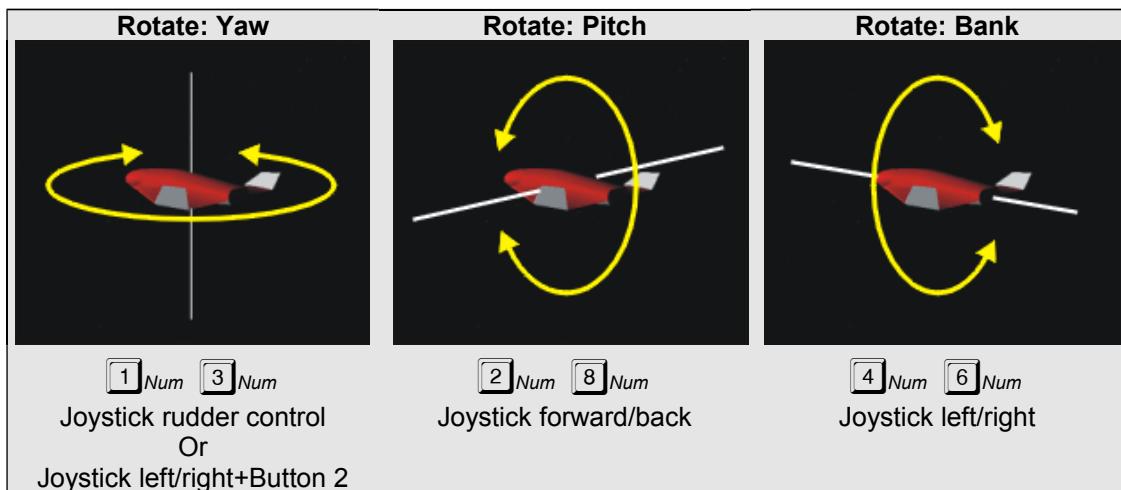
Acceleration from main, retro and hover thrusters

The maximum vacuum thrust ratings for main, retro and hover thrusters as well as the current spacecraft mass are displayed in the vessel's info sheet (**Ctrl** **I**). Values are in Newton ( $1\text{N} = 1\text{kg m s}^{-2}$ ). Note that the actual ratings may be lower in the presence of ambient atmospheric pressure.

## 15.2 Attitude thrusters

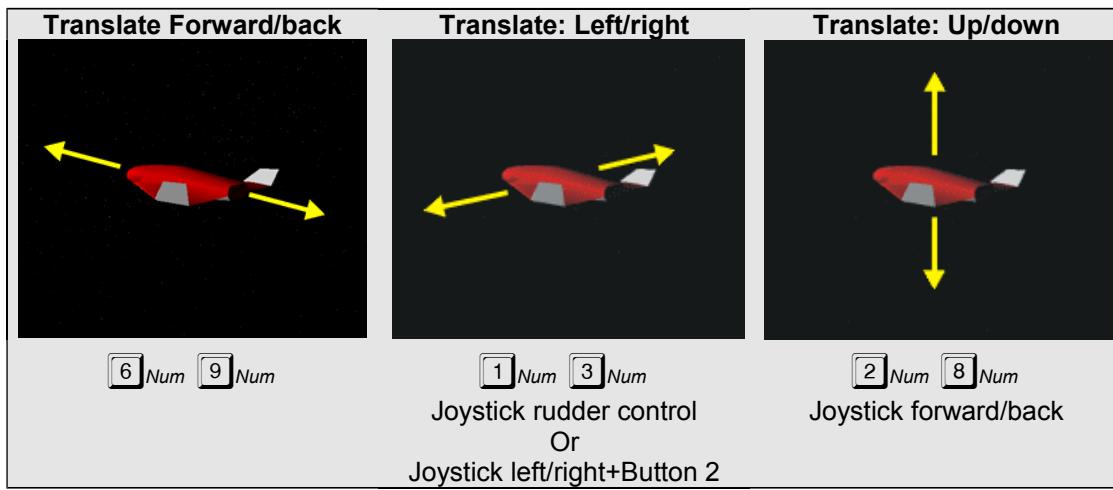
Attitude thrusters are small engines which are engaged in pairs to enable rotation or translation of the spacecraft. In rotation mode, attitude thrusters are fired in cross-linked pairs to produce a rotational moment (e.g. front right and back left to rotate left). In translation mode, thrusters are fired in parallel pairs to produce a linear moment (e.g. front right and back right to accelerate left). The current attitude mode is indicated in the top left corner of the HUD (*Att ROT* and *Att LIN*) and can be toggled with **T** **Num**.

Attitude thrusters are controlled with the joystick or keyboard. In rotation mode:



Attitude thrusters in rotational mode

In translation mode the spacecraft can be linearly accelerated forward/back, left/right and up/down.



*Attitude thrusters in translational (linear) mode*

For fine control of attitude thrusters with the keyboard use **[Ctrl]-Numpad** key combinations. This engages the engines at 10% thrust.

An important control function is the *Kill rotation* sequence (**[5] Num**). This will automatically engage appropriate attitude thrusters to stop the ship's rotation.

## 16 Radio navigation aids

Orbiter uses various types of radio transmitters and receivers to provide information for spacecraft instrument navigation systems. Most vessels are equipped with one or more NAV radio receivers which can be tuned to the frequency of a navigation radio transmitter, and feed the data to the vessel's navigation subsystems.

To tune a NAV receiver, open the Comm Control MFD mode, select a receiver (`Shift` `,` and `Shift` `.`), and tune through the frequency band (`Shift` `[`, `Shift` `]`, `Shift` `-`, `Shift` `=`).

The following types of navaid radio transmitters are currently supported in Orbiter:

- ▶ **VOR:** surface-based omnidirectional radio beacons, typically with a range of several hundred kilometres. VOR signals can be fed into the HSI (horizontal situation indicator) MFD or the VTOL/VOR MFD to obtain direction and distance information. A map with VOR locations is available with `Ctrl` `M`. Frequencies of VOR transmitters located at a surface base are also available from the base's information sheet (`Ctrl` `I`).
- ▶ **VTOL:** Surface landing pads for vertical take-off and landing (VTOL) may be equipped with short-range landing aid transmitters. This signal can be fed to the VTOL/VOR MFD to obtain landing alignment information. A list of available VTOL transmitters can be obtained from the information sheet of a surface base (`Ctrl` `I`).
- ▶ **ILS:** Many runways are equipped with Instrument Landing Systems (ILS) to provide heading and glideslope information. ILS information is used by the HSI MFD mode. ILS frequencies are available from the runway listing in the information sheets of surface bases.
- ▶ **XPDR:** Some spacecraft and orbital stations are equipped with transponders for identification and long-range homing purposes. An XPDR signal can be fed to the Docking MFD to obtain distance and closing speed information. It is also recognised by the Docking HUD mode, which will display a target rectangle, velocity marker and distance information. The Docking HUD can be slaved to a NAV receiver with `Ctrl` `R`. XPDR frequencies can be obtained from a vessel's information sheet (`Ctrl` `I`).
- ▶ **IDS:** Instrument docking system. Most space stations and some spacecraft provide short-range approach signals for their docking ports (typical range 10 km). This signal can be fed to the Docking MFD to obtain dock alignment information. It can also be fed to the Docking HUD to display the approach path as a series of rectangles. IDS frequencies are available from a vessel's information sheet (`Ctrl` `I`).

To find out how to set up XPDR and IDS transmitters via a cfg script see the 3DModel document.

## 17 Basic flight manoeuvres

The following flight techniques are mostly my own invention. They seem plausible, but since I am not a space flight expert (although an enthusiastic amateur) they may be inefficient or plainly wrong. Corrections and suggestions are always welcome.

### 17.1 Surface flight

By *surface flight* I mean flight paths close to a planetary surface which are not actually orbits, i.e. where the gravitational field of the planet must be countered by applying an acceleration vector, rather than the free fall situation of an orbit. Surface-to-surface transfers (from one surface base to another) typically involve surface flight.

#### If the planet has no atmosphere

In this case the only forces acting on your ship are the planet's gravitational field and whatever thrust vectors you apply. Most notably, there is no atmospheric friction to reduce the ship's "airspeed". This causes a flight model rather different from a normal airplane. The simplest, but probably not the most efficient strategy for surface flight is:

- ▶ Use hover thrusters to balance gravitational acceleration (can be done automatically with "Hold altitude" nav mode). This also means the ship should be kept level with the horizon.
- ▶ Navigate with short main thruster bursts.
- ▶ At high horizontal velocities the flight path may approach an orbital trajectory. In that case hover thrusters must be reduced to maintain altitude. In the extreme case of horizontal velocity exceeding the orbital velocity of a circular orbit at zero altitude, the ship will gain altitude even for disengaged hover thrusters. That means you have entered into an elliptic orbit at periapsis.

#### If the planet has an atmosphere

When flying through an atmosphere, the flight model will be similar to an airplane's, in particular if your ship essentially *is* an airplane, i.e. has airfoils that produce a lift vector as a function of airspeed. As with an airplane, you need to apply continuous thrust to counter atmospheric friction and maintain a constant airspeed. If your ship produces lift, hover thrusters are not necessary unless airspeed falls below stall speed (e.g. during vertical lift-off and landing). If your ship does not generate a lift vector, hover thrusters must be substituted, or the ship must be tilted such that the main thrusters provide a vertical component to counter the gravitational field. Note that "lift" produced by thrusters is independent of airspeed.

### 17.2 Launching into orbit

Launching from a planetary surface and entering into a low orbit is one of the most basic problems of space flight. During the early part of the launch the ship needs to apply vertical thrust to overcome the gravitational field and acquire altitude. As the

ship approaches the desired altitude, the pitch is reduced to increase the horizontal acceleration component, in order to reach orbital velocity. A stable orbit is achieved as soon as the periapsis distance is sufficiently high above the planetary surface so that atmospheric friction can be neglected.

Orbits should usually be prograde i.e. rotate in the same direction as the planet surface, to exploit the initial velocity vector provided by the planet. (That is, on Earth ships should be launched eastwards). This also means that launch sites near the equator are most efficient since they provide the largest initial velocity.

#### In Practice:

(This assumes the ship is initially placed on the Earth's surface).

- ▶ Set HUD to surface mode. Bring up Surface and Orbit MFD modes.
- ▶ Engage hover thrusters to at least  $10\text{m/s}^2$ .
- ▶ Once free of the surface, turn towards east ( $90^\circ$  on HUD compass ribbon).
- ▶ Raise nose to  $70^\circ$  pitch, while at the same time engaging full main thrusters.
- ▶ As air speed increases, bring hover thrusters slowly back to zero.
- ▶ As you gain altitude, slowly reduce pitch (e.g.  $60^\circ$  at  $20\text{km}$ ,  $50^\circ$  at  $50\text{km}$ ,  $40^\circ$  at  $80\text{km}$ , etc.)
- ▶ As the desired altitude is reached (e.g.  $200\text{km}$ ) the vertical velocity and acceleration should fall to zero. (by reducing pitch, not by killing the thrusters). Pitch may still be  $> 0$  because part of the thrust vector is required to counter gravitation until full orbital velocity is reached.
- ▶ As the tangential velocity increases, pitch should be reduced to maintain constant altitude.
- ▶ As soon as the tangential velocity for a circular orbit is reached (eccentricity = 0) thrusters should be killed.

### 17.3 Changing the orbit

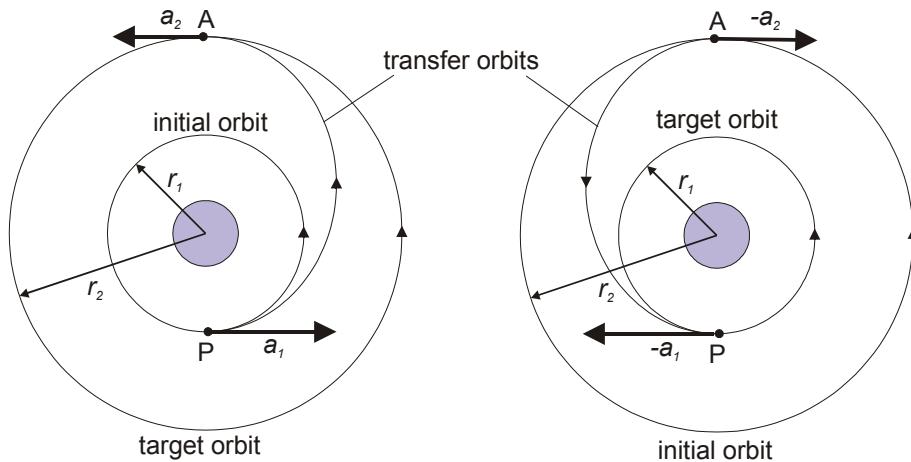
To change the shape of the orbit without changing the orbital plane, the thrust vector must be applied in the orbital plane. The simplest maneuvers involve modifying the apoapsis or periapsis distances.

- ▶ Increase apoapsis distance: Wait until the ship reaches periapsis. Apply thrust vector prograde (ship orientated along velocity vector, engage main thrusters).
- ▶ Decrease apoapsis distance: Wait until the ship reaches periapsis. Apply thrust vector retrograde (ship orientated against velocity vector, engage main thrusters).
- ▶ Increase periapsis distance: Wait until the ship reaches apoapsis. Apply thrust vector prograde.
- ▶ Decrease periapsis distance: Wait until the ship reaches apoapsis. Apply thrust vector retrograde.

### In Practice:

Case 1: Assume you want to change from a low circular orbit (200km) into a higher circular orbit (1000km).

- ▶ Turn ship prograde and engage main thrusters.
- ▶ Kill thrusters as soon as apoapsis distance reaches 1000km+planet radius (e.g. 7370km for Earth). Use *Orbit* MFD mode to monitor this.
- ▶ Wait until you reach apoapsis.
- ▶ Turn ship prograde and engage main thrusters.
- ▶ Kill thrusters as soon as periapsis equals apoapsis and eccentricity is back to 0.



*Moving into a higher orbit involves prograde acceleration at P and A (periapsis and apoapsis of the transfer orbit). Conversely, moving from the higher to the lower orbit requires retrograde acceleration at A and P.*

Case 2: Rotate the argument of periapsis of an elliptic orbit (i.e. rotate the orbital ellipse in its plane).

- ▶ Wait until you reach periapsis.
- ▶ Turn ship retrograde and engage main thrusters until orbit is circular (eccentricity = 0).
- ▶ Wait until you reach the desired new periapsis position.
- ▶ Turn ship prograde and engage main thrusters until original eccentricity and apoapsis distances are re-established.

## 17.4 Rotating the orbital plane

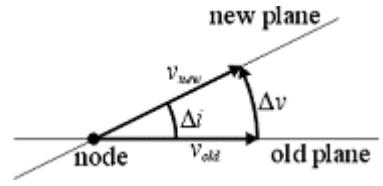
When trying to rendezvous with another object in orbit, the required orbit changes can often be simplified by splitting them into two separate phases: a plane change that rotates the plane of the current orbit into that of the target, and further in-plane operations that only require the application of thrust in the plane of the orbit. Once you are in the same plane as your target, most of the following navigational problems

become essentially two-dimensional, which makes them more robust and a lot easier to compute.

In terms of the orbital elements, aligning the plane of the orbit with a target plane means to match the two elements which define the orientation of the orbit in space: inclination ( $i$ ) and longitude of the ascending node ( $\Omega$ ).

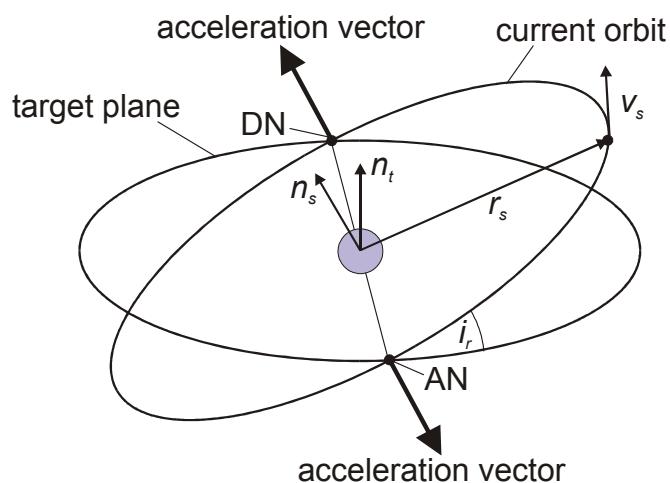
The rotation of the orbital plane requires the application of out-of-plane thrust. To match the plane with a target plane, thrust should be applied normal to the current plane, in one of the nodes (the points where the orbit crosses the intersection of the current and target planes). This will rotate the orbital plane around an axis defined by your current radius vector.

The amount of normal  $\Delta v$  required to rotate by a given angle  $\Delta i$  is proportional to the orbital velocity  $v$ . It is therefore more fuel-efficient to perform the plane change where  $v$  is small, i.e. close to aphelion. For a given line of nodes, it is more efficient to perform the plane change at the node closer to aphelion. Sometimes it may even be useful to make the orbit more eccentric prior to the plane change maneuver, so that the radius distance of one of the nodes is increased.



#### Note:

- ▶ If the angle between the initial and target OP is large it may be necessary to adjust the orientation of the spacecraft during the maneuver to keep it normal to the OP.
- ▶ It may not be possible to align the plane in a single node crossing. If the angle towards the target plane cannot be reduced further by accelerating normal to the current orbit, cut the engines and wait for the next node crossing.
- ▶ Since the maneuver will take a finite amount of time  $\Delta T$ , thrusters should be engaged approximately  $1/2 \Delta T$  before intercepting the node.



*Alignment of the orbital plane.  $r_s$ : radius vector.  $v_s$ : velocity vector. AN: ascending node. DN: Descending node.  $n_s$ : normal of the current plane.  $n_t$ : normal of the target plane.*

The direction of the normal vector  $n_s$  is defined by the direction of the cross product  $r_s \times v_s$ . Acceleration should be applied in direction  $-n_s$  in the ascending node (AN), and in direction  $+n_s$  in the descending node(DN). (see Error: Reference source not found).

### In Practice:

- ▶ The *Align orbital plane* MFD mode (see Section 14.8) is designed to aid in plane alignment. Select the target object ( ).
- ▶ The HUD should be in Orbit mode. As your ship approaches the intersection with the target plane, rotate it to a normal (if at DN) or anti-normal (if at AN) orientation to the current orbital plane. There are automated RCS sequences ( and ) available to perform the required alignment. Use the HUD Orbit inclination ladder to monitor progress.
- ▶ As soon as the time to node ( $T_n$ ) reaches half the estimated burn time ( $T_{thA}$  or  $T_{thD}$  for AN and DN, respectively) the “Engage thruster” indicator will start flashing. Engage full main thrusters. Make sure the relative inclination ( $RInc$ ) decreases, i.e. the rate of change ( $Rate$ ) is negative, otherwise you may be pointing in the wrong direction.
- ▶ Adjust the ship’s orientation as required to keep normal to orbital plane (the automated RCS sequences will do this for you).
- ▶ Disengage thrusters as soon as the action indicator turns back to “Kill thruster”.
- ▶ If the relative inclination was not sufficiently reduced repeat the procedure at the next node passage.
- ▶ During the maneuver make sure your orbit does not become unstable. Watch in particular for the eccentricity (use the Orbit MFD to monitor this).

## 17.5 Synchronising orbits

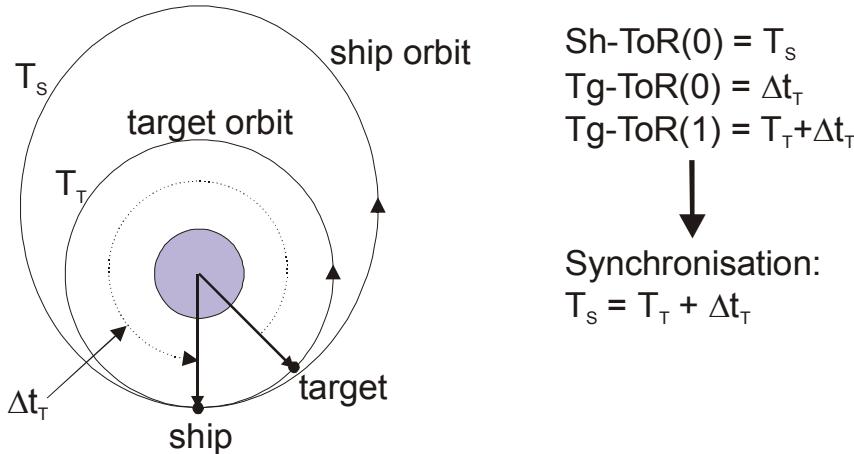
This section assumes that the orbital planes of ship and target have been aligned (see previous section).

The next step in a rendezvous maneuver after aligning the orbital planes is to modify the orbit in the plane such that it intercepts the target’s orbit and both ship and target arrive simultaneously at the interception point. Use the *Synchronise Orbit* MFD to calculate the appropriate orbit.

For simplicity we first assume that the ship and target are in a circular orbit with the same orbital radius (for synchronising the orbital radius see Section 17.3), i.e. both objects have the same orbital elements except for the mean anomaly. The method for intercepting the target is then as follows:

- ▶ Switch the reference mode of the *Synchronise Orbit* MFD to “Manual” and rotate the axis to your current position.
- ▶ Turn your ship prograde (using Orbit HUD mode) and fire main thrusters.

- ▶ The orbit will become elliptic, with increasing apoapsis distance. Periapsis is your current position. Simultaneously the orbit period and the times to reference axis will increase.
- ▶ Kill thrusters as soon as one of the Sh-ToR times coincides with one of the Tg-ToR times.
- ▶ Then you just have to wait until you intercept the target at the reference axis.
- ▶ At interception, fire thrusters retrograde to get back to the circular orbit and match velocity with the target.



*A transition orbit to intercept the target at the next periapsis passage.*

#### Notes:

- ▶ Instead of increasing the apoapsis distance one could fire retrograde and reduce the periapsis distance in this maneuver. This may be more efficient if the target is ahead of the ship. But make sure that periapsis does not become dangerously low!
- ▶ It should always be possible to match your next ToR (orbit 0) with the target's ToR at orbit 1. If you are low on fuel it may however be better to match later orbits if this can be achieved with less distortion to the original orbit. For example, if the target is marginally ahead, then to intercept it in the next orbit you need to nearly double your orbital period.
- ▶ It is not essential that the orbits are identical or circular at the start of the maneuver. It is sufficient for them to intersect. In that case it is best to use Intersection 1 or 2 reference mode in the *Synchronise* MFD.
- ▶ You don't necessarily need to wait until you reach the reference point before firing thrusters, but it simplifies matters because otherwise the intersection point itself will move, making the alignment of orbit timings more difficult.

## 17.6 Landing (runway approach)

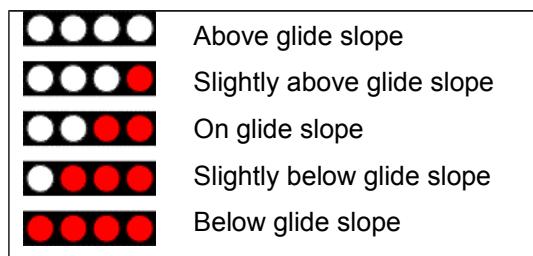
Some of Orbiter's spacecraft support powered or unpowered runway approaches, similar to normal aircraft. Examples are the delta glider and the Space Shuttle. The Shuttle Landing Facility (SLF) at the Kennedy Space Center provides an good opportunity for exercising landing approaches.

## Visual approach indicators

The visual approach aids at the SLF are designed for Shuttle landings. They include a Precision Approach Path Indicator (PAPI) for long-range glide slope alignment, and a Visual Approach Slope Indicator (VASI) for short-range alignment. The PAPI is set up for a glide slope of  $20^\circ$  (about 6 times as steep as standard aircraft approach slopes!). The VASI is set up for a  $1.5^\circ$  slope during the final flare up prior to touch-down.

### Precision Approach Path Indicator

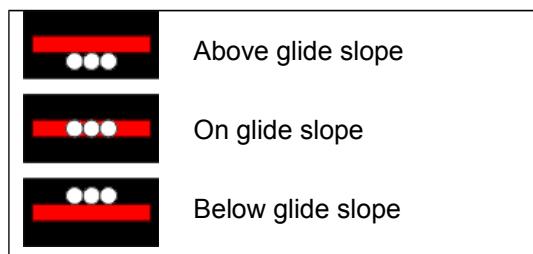
The PAPI consists of an array of 4 lights, which appear white or red to the pilot depending on his position above or below the glide slope. At the correct slope there will be 2 white and 2 red lights (see figure). In Orbiter there are 2 PAPI units per approach direction at the SLF, located about 2000 meters in front of the runway threshold.



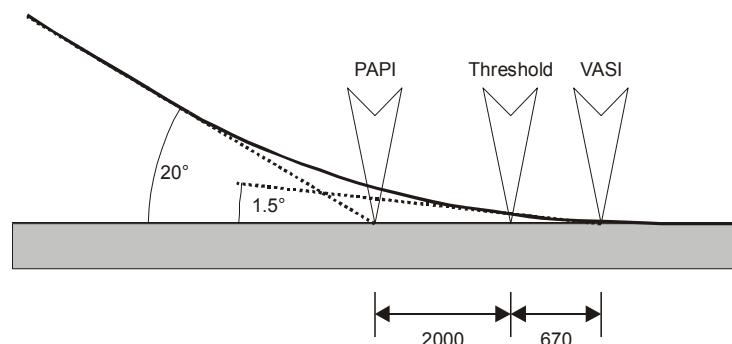
*PAPI indicator signals*

### Visual Approach Slope Indicator

The VASI consists of a red bar of lights, and a set of white lights in front of them. At the correct slope, the white lights are aligned with the red bar. (see figure). At the SLF, the VASI is located about 670 meters behind the runway threshold.



*VASI indicator signals*



## 17.7 Docking

Docking to an orbital station is the last step in the rendezvous maneuvered. Assuming you have intercepted the target station following the preceding steps, here we discuss the final docking approach.

- ▶ Select *Docking mode* in one of your MFD displays, and the *Docking* HUD by pressing **H** until docking mode is selected.
- ▶ Tune one of your NAV receivers to the station's XPDR frequency, if available. The frequency is listed in the station's information sheet (**Ctrl I**).
- ▶ Slave the Docking MFD and Docking HUD to that NAV receiver (**Shift N** and **Ctrl R**, respectively)
- ▶ If not done already, synchronise relative velocity by turning the ship until it is aligned with the relative velocity marker (⊕) and fire main thrusters until velocity value (V) approaches zero.
- ▶ Rotate the ship to face the station (⊖ marker).
- ▶ At a range of approx. 10 km, tune a NAV receiver to the IDS (Instrument Docking System) frequency of the designated docking port, if available. Slave Docking MFD and Docking HUD to that receiver, if applicable. This will display orientation and direction information in the MFD, and a visual representation of the approach path in the HUD (rectangles).
- ▶ Move towards the approach path rectangle furthest away from the station and hold.
- ▶ Align the ship's heading with the flight path direction using the 'X' indicator in the MFD.
- ▶ Align the ship's position on the approach path using the '+' indicator in the MFD. Switch attitude thrusters to linear mode for this.
- ▶ Align the ship's rotation along its longitudinal axis using the arrow indicator in the MFD.
- ▶ Approach the station by engaging main thrusters briefly. During approach correct your position continuously using linear attitude thrusters.
- ▶ Slow down approach speed to less than 0.1m/s before intercepting the dock.
- ▶ You need to approach the dock to less than 0.3 m for a successful docking maneuver.
- ▶ To disengage from the docking port, press **Ctrl D**.

*A Shuttle-A class cargo ship after successfully completed docking approach to the ISS.*

**Notes:**

- ▶ For precise attitude control with the keyboard use the attitude thrusters in “low power” mode (Ctrl + Numpad key).
- ▶ Rotational alignment is not currently enforced, but may be in future versions.
- ▶ Currently no collision tests are performed, so you might fly straight through the station if you miss the docking approach.

**Docking at rotating stations**

Stations like Luna-OB1 rotate to use centrifugal forces for emulating gravity – which is nice for their inhabitants but makes docking a bit more complicated. Docking is only possible along the rotation axis, so at most 2 docking ports can be provided. The docking procedure is similar to the standard one, but once aligned with the approach path, the rotation around the ship’s longitudinal axis must be aligned with that of the station.

**Important:**

- ▶ Initiate your ship’s longitudinal rotation only immediately before docking (when past the last approach marker). Once you are rotating, linear adjustments become very difficult.
- ▶ Once the rotation is matched with the station, don’t hit  $\boxed{5}_{Num}$  (Kill rotation) by accident, or you will have to start the rotation alignment again.

**Cheat:**

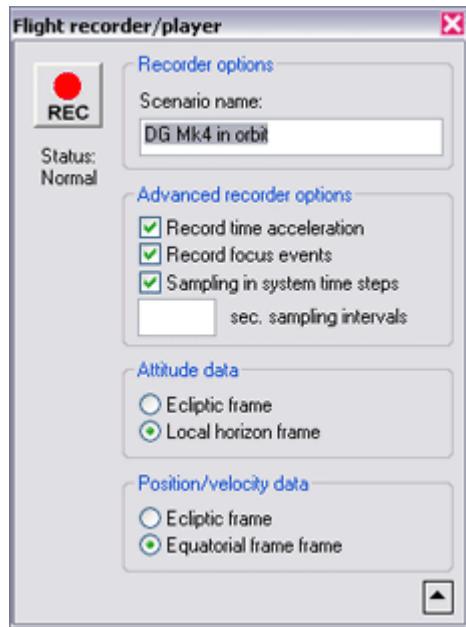
Since rotational alignment is not enforced at present, you can simply ignore the rotation of the station and fly straight in.

## 18 Flight recorder

You can record and play back your Orbiter simulation sessions with the built-in flight recorder feature. To access the recorder during a simulation, open the *Flight recorder/player* dialog with **Ctrl** **F5**. You can now select a name for the recorded scenario. By default, the recording will be stored under the current scenario name. Then press the *REC* button to begin recording the flight to disc. Press the *STOP* button to turn the recorder off again. You can also start and stop the recorder directly from the simulation with the keyboard shortcut **Ctrl** **C**. An active recorder is indicated by a “Record” box in the simulation window.

Some additional recorder options can be accessed by pressing the  $\nabla$  button. These include:

- ▶ **Record time acceleration:** this option records any changes in time compression during the recording. During playback, the user has then the option to set time compression automatically from the recorded data.
- ▶ **Sampling in system time steps:** If ticked, the intervals between recorded data samples are determined in system time, otherwise in simulation time. In system time, sampling is less dense during time compression. This allows to reduce the size of the data files when recording over long periods of time, by fast-forwarding through less critical parts of a mission.
- ▶ **Sampling intervals:** Currently not used.
- ▶ **Attitude data:** Can be recorded either with respect to the global ecliptic frame of reference, or with respect to the local horizon of the current reference celestial body.



To play back a previously recorded session, launch the scenario under the *Playback* scenario folder. During playback, all vessels will follow their pre-recorded trajectories and will not respond to manual user control. At the end of the playback, the simulation will automatically switch back to manual mode, and the user can take over control. You can terminate the playback before the end of the recorded data is reached by pressing **Ctrl** **C**, or by pressing the *STOP* button in the Recorder dialog. In that case, control returns immediately to the user.

During playback, the user has still various options to interact with the simulation. For example, it is possible to move the camera, change between internal and external views,



and even manipulate the MFD instruments to access more flight data. Manual control of time compression is only possible if the *Play at recording speed* option is deactivated in the Player dialog ( ). Otherwise, Orbiter sets the time compression directly from the recorded data.

Recorded flights can be annotated with onscreen notes which appear on the simulation window at predefined times. This opens an exciting new way to write tutorials and space flight demonstrations. (The annotations can be turned off from the recorder dialog during a playback by deactivating the *Show inflight notes* option).

The data for recorded simulation sessions are stored under the *Flights* subdirectory. Orbiter creates a new folder for each recording, using the same name as the recorded scenario. Each vessel in the scenario writes three data streams to this folder, including:

- ▶ **position and velocity** (\*.pos). At the moment, these data are recorded relative to the reference planet, either in a non-rotating reference system (ecliptic and equinox of J2000), or a rotating equatorial reference system. As a result, trajectories are recorded in an absolute time frame. Samples are written in regular intervals (currently 4 seconds) or if the velocity vector rotates by more than 5 degrees.
- ▶ **attitude** (\*.att). Attitude data are saved in terms of the Euler angles of the spacecraft with respect to the ecliptic reference frame or local horizon frame. Samples are written whenever one of the angles has changed by more than a predefined threshold limit.
- ▶ **articulation events** (\*.atc). This stream contains changes in thrust levels of spacecraft engines, and other types of events (e.g. change of RCS mode, activation/deactivation of navigation modes, etc). It can also be used by individual vessel modules to record custom events such as animations. Further, time compression events can be recorded here. This data stream can also contain timed annotations that are displayed on top of the simulation window during replay. Annotations must be added manually to the atc file after the recording is completed.

A complete recording session consists of the playback scenario in the Scenarios\playback folder, and the corresponding flight data folder under the Flights directory. To share a playback with other Orbiter users, these files must be copied. Note that the scenario file can be moved to a different Scenario folder, but no two playback scenarios can have the same name.

Be aware that long simulation sessions (in particular during time acceleration) may lead to very large data files if the *Sampling in system time steps* option is not used.

Note that the recorder feature is still under development. Future versions may introduce changes to the recording mechanism and file formats. Certain features, such as the recording of animations, require modifications to the vessel plugin modules and may not be available for all vessel types.

Implementation details and flight recorder file format specifications can be found in a separate Orbiter Technical Note [Doc\Technotes\RecorderRef](#).

## 18.1 Playback event editor

After recording a flight, you have the option of using the *Playback editor* to enhance the playback. You can

- ▶ Add annotations which appear on the simulation window at particular times during the replay.
- ▶ Change camera positions.
- ▶ Modify the time compression applied during the playback

However, you can not modify the recorded simulation itself (such as changing vessel trajectories, engine burn times, animations, etc.) To access the Playback editor, start a previously recorded scenario, then open the playback dialog with **Ctrl F5**, and click the *Playback editor* button. This will open the playback editor dialog box.

### Event list

The top part of the dialog box contains the *event list* for the scenario. Initially, this may be empty, or contain some event tags that were inserted during the original recording.

Each line of the list represents an event. Each event contains

- ▶ a time stamp (simulation time since the start of the playback in seconds)
- ▶ an event tag defining the type of event
- ▶ and event-type specific parameters.

The current simulation time is indicated by a blinking line (< < < <). As the playback progresses, the current time marker is moving through the event list. Sometimes it may be useful to pause a playback (**Ctrl P** in the simulation window) to have more time for editing the event list.

### Adding a new event

Events are always added at the current simulation time (that is, at the position of the time marker). You can however later edit the time stamps of events to move them to a different time.

First, select an event type from the drop-down list below the event list. The most important event types have entries in the list. (For less frequently used items, select the entry and enter the event tag manually). Then press the *Insert* button.

A new event line will appear in the list, and the *Edit* area in the lower part of the dialog box will allow to define any required parameters.

### Editing an existing event

If you want to modify the parameters of an event already in the list, simply highlight it by clicking on a line in the list. The event parameters will appear in the *Edit* area, and you can modify them.

### Deleting events

To delete an event, highlight it by clicking on a line in the list. Then press the *Delete* button.

### **Committing changes**

To commit the changes you have made to the event list, press the *Commit* button at the bottom of the dialog box. This will save the modified event list in the playback file. Orbiter will immediately re-scan the file up to the current simulation time, so that any changes can be examined at once.

## NEW! Script interface

Orbiter contains a script interpreter module which allows to control a variety of simulation tasks with the help of scripts. Script applications include autopilots, MFD control, interactive tutorials, mission control, and many others.

The Orbiter script engine uses the Lua scripting language ([www.lua.org](http://www.lua.org)). A large number of functions and methods have been added to the standard Lua command set to provide an interface to the Orbiter simulation environment. To a large extent the Lua-Orbiter interface replicates the Orbiter C++ API interface.

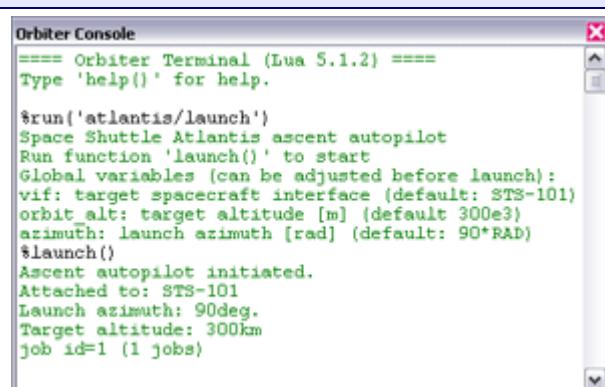
There are several methods available to users and developers for accessing the script interface:

- ▶ Console window: Make sure the *LuaConsole* module is activated in the Modules tab of the Orbiter launchpad. The console can then be opened by selecting *Lua console window* from the custom command list (**Ctrl F4**).
- ▶ Terminal MFD: Make sure the *LuaMFD* module is activated in the Modules tab of the Orbiter launchpad. The terminal MFD mode is then available via **Shift F1** - **Shift T**.
- ▶ Run a script on launching a scenario. This is useful for mission- or tutorial-style scenarios.
- ▶ Execute a Lua command or script from a module using the Orbiter API. This is useful for implementing autopilots and control systems.

For script examples and a list of available functions, see the “Orbiter Scripting User Manual” section in the Orbiter online help (available from the “Help” button on the Launchpad window, or with **Alt H** from within the simulation. From a terminal (console window or terminal MFD) you can access the script manual by typing: *help(api)*.

### 19.1 Console window

The console allows to enter commands or launch scripts controlling various aspects of spacecraft behavior. To open the console window, select *Lua console window* from the custom command list (**Ctrl F4**). If this option is not available, activate the *LuaConsole* module in the Modules tab of the Orbiter launchpad dialog.



The screenshot shows the Orbiter Console window titled "Orbiter Console". The window displays the following text:  
==== Orbiter Terminal (Lua 5.1.2) ====  
Type 'help()' for help.  
  
\$run('atlantis/launch')  
Space Shuttle Atlantis ascent autopilot  
Run function 'launch()' to start  
Global variables (can be adjusted before launch):  
vif: target spacecraft interface (default: STS-101)  
orbit alt: target altitude [m] (default 300e3)  
azimuth: launch azimuth [rad] (default: 90\*RAD)  
#launch()  
Ascent autopilot initiated.  
Attached to: STS-101  
Launch azimuth: 90deg.  
Target altitude: 300km  
job id=1 (1 jobs)

The console window is a simple terminal interface. User input is shown in black, program responses are shown in green. The window can be resized. The font size

can be selected via the *Console configuration* item in the Extras tab of the Orbiter launchpad. The console allows simple command line editing, and scrolling through the command history via the **[↑]** key.

## 19.2 Terminal MFD

The Terminal MFD mode is available via the *Terminal MFD* entry (**[Shift][T]**) of the MFD selection list (**[Shift][F1]**). If this entry doesn't isn't available, activate the *Luamfd* module in the Modules tab of the Orbiter launchpad.

Commands can be entered into the MFD by pressing the INP (input) button (**[Shift][I]**), typing the command, and pressing **[Enter]**.

The MFD allows to open multiple command interpreters simultaneously. To open a new terminal page, press NEW (**[Shift][N]**). To switch between pages, press PG> (**[Shift][.]**) or <PG (**[Shift][,]**). To close a terminal page, press DEL (**[Shift][D]**).

## 19.3 Run a script with a scenario

To run an Orbiter script automatically when a scenario starts, the scenario must contain the following line inside the ENVIRONMENT block:

```
SCRIPT <path>
```

where *<path>* is the path to the script file, relative to the Script subdirectory. Script files are text files. The file names should have the extension ‘.lua’, but the extension should not be added to the path specification. For example, if you have created a script file tutorial1.lua in a directory Script\MyScripts under the Orbiter main directory, the entry in the scenario file would be

```
SCRIPT MyScripts/tutorial1
```

## 19.4 Call a command or script via the API

To access the script interface from within a plugin module, you must create an interpreter instance with the *oapiCreateInterpreter* function. This returns a handle which can subsequently be used to issue commands via the *oapiExecScriptCmd* function. It is possible to either execute individual commands, or entire scripts via the ‘run’ command.

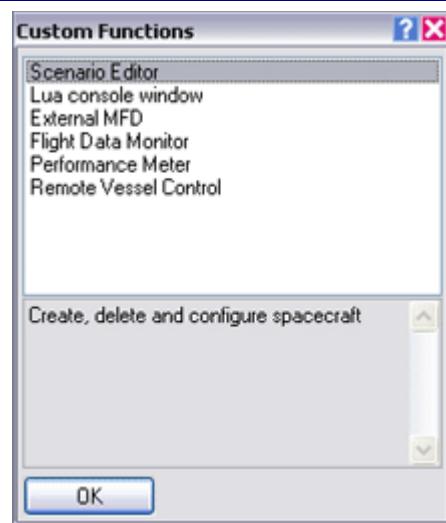
## 20 Extra functionality

Orbiter comes with a default set of plugin modules to enhance the core functionality of the simulator. To access these additional functions, the appropriate modules must be loaded in the Modules tab of the Orbiter Launchpad dialog (see Section 4.4 on how to activate plugin modules).

Many more plugins are available from 3<sup>rd</sup> party addon developers. Check out the Orbiter repositories on the web to find more.

You should only activate modules you want to use, because many plugins may access the CPU even if they are running in the background. Too many active modules can degrade simulation performance.

When activated, some plugins, such as custom MFD modes, take effect automatically whenever the simulation runs. Others are accessible via the Custom functions dialog. Press **Ctrl F4** to get a list of the available functions.



### 20.1 Scenario editor

Orbiter has an editor that allows to create, configure and delete vessels within a running simulation, and to change the simulation time. The editor is provided as a plugin module. To use it, make sure that the *ScnEditor* module is activated in the *Modules* tab of the Orbiter Launchpad dialog.

During the simulation, you can access the editor by opening the *Custom Functions* dialog with **Ctrl F4**, and double-clicking the *Scenario Editor* entry in the list. This will bring up the editor's main page. From here, you can either configure any vessels currently present in the simulation, or create new vessels in any location.

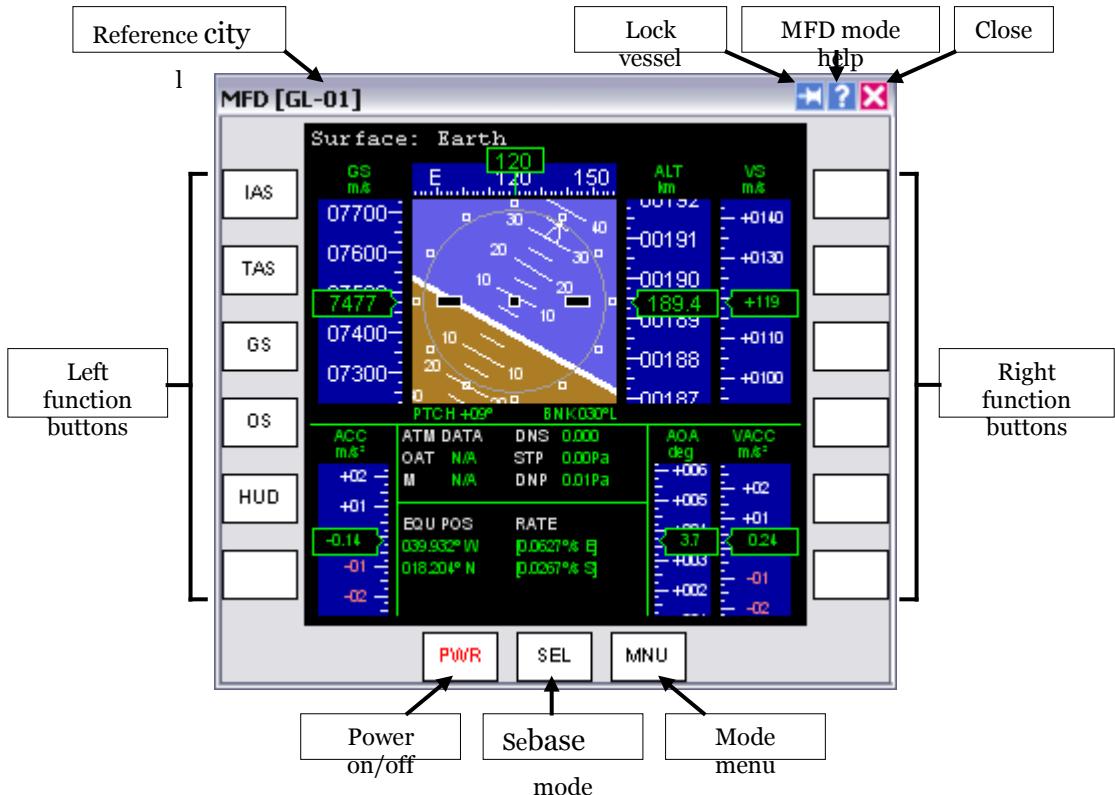
The operation of the scenario editor is described in a separate document: [Doc\ScenarioEditor](#). This also contains a section for vessel addon developers who want to integrate the scenario editor with their vessel code.

### 20.2 External MFDs

If the multifunctional displays (MFD) integrated in the vessel instrument panels don't provide enough information, you can open additional MFD displays in external windows. This is particularly useful in multi-monitor setups where you can display the Orbiter simulation window on one monitor, and a set of MFDs on the other.

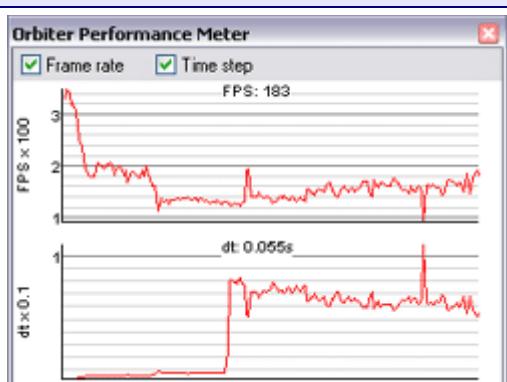
To open external MFDs, the *ExtMFD* module must be activated in the Orbiter Launchpad dialog. You can then open any number of MFD windows by clicking *External MFD* from the Custom Functions dialog (**Ctrl F4**).

External MFDs behave in the same way as built-in MFDs. They can be controlled by pressing the buttons on the left, right and bottom edges. See Section 14 for a description of the available MFD modes and controls.



## 20.3 Performance meter

This is a little dialog box to keep track of Orbiter's frame rate performance and simulation time step intervals. It shows the frames per second (FPS) and/or the step length interval (in seconds) between consecutive frames in a graphical display over the last 200 seconds. This is a useful tool to estimate the impact of complex scenery and visual effects on the simulation performance. The time step graph also incorporates the effect of time acceleration, and thus reflects the fidelity of the physical model (accuracy of trajectory calculation, etc.)



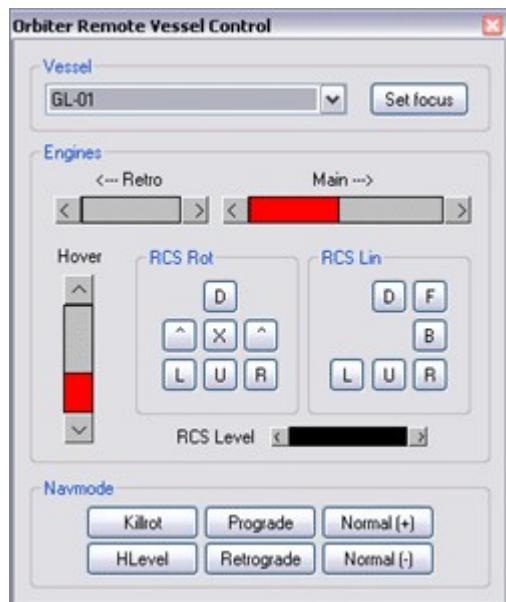
This function is only available if the *Framerate* module is active and is accessible via the *Frame Rate* entry in the *Custom functions* panel (**Ctrl F4**).

## 20.4 Remote vessel control

The *Remote Vessel Control* plugin allows to remotely control the engines of all spacecraft.

The dialog contains the a vessel selection list, gauges for main, retro and hover engines, controls for RCS thrusters in rotational and linear mode, and access to the standard navmode functions. This interface can also be useful if simultaneous access to linear and rotational RCS modes is required.

This tool is available only if the *Rcontrol* module is active, and can be accessed via the *Remote Vessel Control* entry in the *Custom functions* panel (**Ctrl F4**).



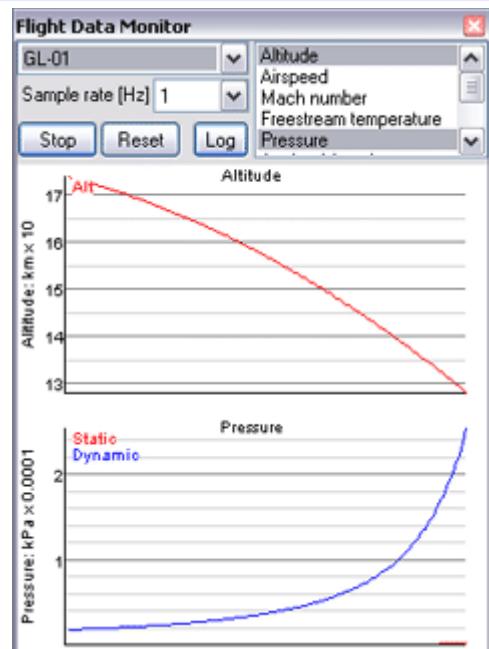
## 20.5 Flight data monitor

The flight data monitor graphically displays a number of flight parameters as a function of time. This tool is available only if the Flight-Data module is active. The dialog box is accessible via the *Custom functions* panel (**Ctrl F4**).

The control area of the dialog box allows to select the vessel for which the flight data are displayed, the sampling rate, and the flight parameters to show.

The following parameter displays are currently supported:

- ▶ Altitude
- ▶ Airspeed
- ▶ Mach number
- ▶ Free stream temperature
- ▶ Static and dynamic pressure
- ▶ Angle of attack
- ▶ Lift and drag force
- ▶ Lift over drag ratio (L/D)
- ▶ Vessel mass



For each parameter category selected in the list, a graph display is opened below the control area to track that parameter as a function of time.

- ▶ The Start/Stop button starts or stops the update of the data graphs.
- ▶ The Reset button clears the data graphs.
- ▶ The Log button starts or stops the output of flight data to a log file. When the Log button is ticked, Orbiter will write out data into text file *FlightData.log* in the main Orbiter directory. This file can later be used to analyse or visualise the data with external tools. *FlightData.log* is overwritten whenever Orbiter is restarted.

## 21 Flight checklists

This section contains point-by-point checklists for some complete flights. While flying these checklists, you may want to save regularly ( $\text{Ctrl}\text{S}$ ), so you can pick up from a previous state if necessary.

The checklists can also be accessed during the simulation when running a checklist scenario by calling up help ( $\text{Alt}\text{F1}$ ) and clicking the *Scenario* button in the help window. Other scenarios may also provide online help.

### 21.1 Mission 1: Delta-glider to ISS

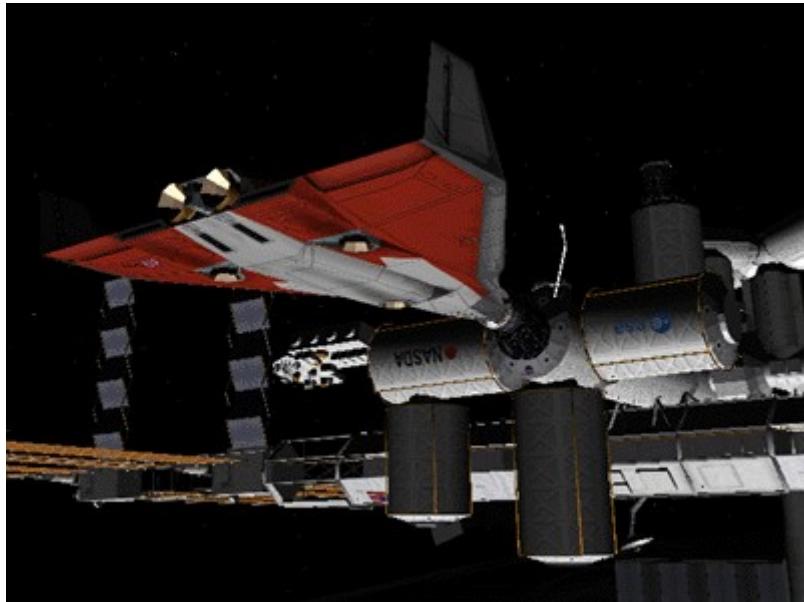
In this mission we launch the Delta-glider into orbit from runway 33 of the Shuttle Landing Facility (SLF) at the Kennedy Space Center, and perform a rendezvous and docking maneuver with the International Space Station.

- ▶ Start Orbiter with the *Checklists|DG to ISS* scenario. Your glider is ready for takeoff from SLF runway 33 at the KSC.
- ▶ You may need to scroll the instrument panel down a bit ( $\uparrow_{cur}$ ) to see the runway in front of you. Make sure you can still see the top half of the panel with the MFD screens.
- ▶ Your launch is scheduled at MJD=51983.6308 (the *Modified Julian Date*, or MJD, is Orbiter's universal time reference, and is shown in the top right corner of the screen). This leaves plenty of time to get used to the instrumentation. If you are not yet familiar with the glider's panel layout, check section 10.1. For details on MFD modes, see section 14.
- ▶ The left MFD screen is in *Surface* mode and shows velocity and altitude data.
- ▶ The right MFD screen is in *Map* mode and shows your current location (KSC) as a white cross. The orbital plane of the ISS is shown as a yellow curve. As time progresses, the curve will shift across the map, as the Earth rotates under the station's orbital plane.
- ▶ To fast-forward to your launch window, press  $\text{T}$ . (Each time you press  $\text{T}$ , time accelerates by a factor of 10). As you approach launch time, switch back to real-time by pressing  $\text{R}$  until the "Wrp" indicator in the top right corner of the screen disappears.
- ▶ Engage main engines ( $\text{Ctrl}+\text{Num}$ ) to 100% thrust. You may also use the sliders on the instrument panels or the throttle control on your joystick to operate the main engines.
- ▶ At ground speed 100 m/s (surface MFD or HUD readout), pull the stick (or press  $\text{2}_{Num}$ ) to rotate.
- ▶ Climb at 10° and retract the landing gear ( $\text{G}$ ).
- ▶ Turn right towards heading 140°.
- ▶ Pitch up steeply to 70°.

- ▶ At about 30km altitude your glider will start to drop its nose due to decreasing atmospheric pressure, even while you are pulling back on the stick. Now activate the RCS (Reaction Control System) by right-clicking the “RCS Mode” selector (on the right side of the instrument panel) or by pressing **Ctrl** **/** **Num**. You are now controlling your craft with attitude thrusters.
- ▶ Pitch down to about  $20^\circ$ . After leaving the dense part of the atmosphere, you need to gain tangential velocity to achieve orbit. Your flight path indicator (the “ $\oplus$ ” symbol on the HUD) should stay above  $0^\circ$ .
- ▶ Switch the right MFD to Orbit mode (*SEL, Orbit*). Select the ship’s orbit as reference plane (**Shift** **P**) and select ISS as target (**Shift** **T**, ‘ISS’).
- ▶ Continue at 100% thrust. Maintain your heading, and adjust pitch angle so that the flight path vector remains slightly above  $0^\circ$ . You will see how your orbit trajectory (green curve in the Orbit MFD) grows.
- ▶ Cut thrusters when your apogee radius (highest point of the orbit) reaches 6.731M (the “ApR” entry in the left column of the Orbit MFD). This corresponds to an altitude of 360 km.
- ▶ Switch to Orbit HUD mode (**H**, **H**)
- ▶ So far we are on a ballistic flight path that would eventually bring us back to the surface. To enter orbit, we need to perform a further burn (“orbit insertion burn”) at the apex of the trajectory. Wait until you reach apogee (the remaining time is shown in the “ApT” entry of the Orbit MFD). This could take a while, so you may want to time-accelerate.
- ▶ At apogee, press the “Prograde” button to turn prograde. Once the velocity marker (“ $\oplus$ ”) is centered on the screen, engage main thrusters until orbit eccentricity (“Ecc”) reaches a minimum, and perigee radius (“PeR”) equals ApR. (this will require only a short burn!)
- ▶ Switch the left MFD to Align Orbital Plane (*SEL, Align planes*). Select ISS (**Shift** **T**, ‘ISS’).
- ▶ Ideally the orbital planes should already be roughly aligned (RInc within  $5^\circ$ ). You now need to fine-adjust the plane.
- ▶ As your ship (P) approaches an intersection point with the target plane (AN or DN): Rotate the ship perpendicular to your current orbital plane ( $90^\circ$  on the Orbit HUD inclination ladder). If you are approaching the ascending node (AN), turn *orbit-antinormal*. If you are approaching the descending node (DN), turn *orbit-normal*. You can use the Auto-navigation modes (**:** for orbit-normal and **.** for orbit-antinormal) to obtain the correct orientation.
- ▶ As soon as the “Engage engines” indicator starts flashing, engage full main engines. The relative inclination between the orbital planes should now decrease.
- ▶ Kill thrusters as soon as the “Kill thrust” indicator appears. If you could not reduce the orbit inclination sufficiently (within  $0.5^\circ$ ) repeat the process at the next nodal point.

- ▶ Once the planes are aligned, the next step is intercepting the ISS. Switch to Sync Orbit MFD (*SEL, Sync orbit*). Switch the reference point to “Intersect 1” or “Intersect 2” “**(Shift M)**”. If the orbits don’t intersect, select “Sh periapsis” instead.
- ▶ The two columns on the right of the MFD screen show the times it will take you (Sh-ToR) and your target (Tg-ToR) to pass the reference position at your current orbit (Ob o) and the 4 subsequent orbits (Ob 1-4).
- ▶ Turn the ship prograde (align with “⊕” velocity marker of the orbit HUD). This can be done by engaging the Prograde auto-navigation mode (**[I]**).
- ▶ Fire main engines until Sh-ToR(o) matches Tg-ToR(1). You will now intercept the ISS at your next passage of the reference point. You may want to engage time acceleration until the time-on-target counters are close to zero, indicating that you are approaching the encounter point.
- ▶ On approach, tune your NAV receivers to the station’s navaid radio transmitters: Select Comm MFD mode (*SEL, COM/NAV*), and tune NAV1 to 131.30 MHz (ISS XPDR frequency) and NAV2 to 137.40 MHz (Dock 1 IDS frequency).
- ▶ Switch to Docking HUD mode (**[H][H]**) and to Docking MFD (*SEL, Docking*).
- ▶ Make sure both HUD and Docking MFD are slaved to NAV1 (use **[Ctrl R]** to cycle through the NAV receivers for the HUD, and **[Shift N]** for the MFD).
- ▶ Rotate the ship to align with the HUD relative velocity marker (“⊕”) and fire main engines until relative velocity is close to zero.
- ▶ Rotate the ship towards the ISS (≤ target designator box) and move to within 5km of the station. You may want to use attitude thrusters in linear (translatorial) mode for this. Switch between linear and rotational mode with the **[L] Num** key.
- ▶ Slave HUD and Docking MFD to NAV2. If you are within 10 km of the ISS you will receive the signal of the IDS system for dock 1, providing alignment information in the MFD and a visual representation (series of rectangles) of the approach path on the HUD.
- ▶ Move towards the rectangle furthest away from the station and hold.
- ▶ Align your ship’s longitudinal axis with the approach path direction (align “X” indicator in the MFD) using attitude thrusters in rotational mode.
- ▶ Align your ship’s rotation around its longitudinal axis (align “^” indicator at 12 o’clock position in the MFD).
- ▶ Center your ship on the approach path (align “+” indicator in the MFD) using linear attitude thrusters.
- ▶ Expose the docking mechanism under the nose cone by pressing **[K]**.
- ▶ Start moving towards the dock with a short burst of the main engines. Closing speed (CVel) should be gradually reduced as you approach the dock. Final speed should be < 0.1 m/s. Re-align ship on the approach path with linear attitude thrusters as required.

- ▶ The docking mechanism should engage once you are within 0.3 m of the designated dock. A “Dock” indicator will appear in the MFD once your ship has successfully docked.
- ▶ Finished!



Mission 1 completed successfully.

## 21.2 Mission 2: ISS to Mir transfer

This mission performs an orbital transfer from the International Space Station to the Russian Mir station (which in Orbiter’s virtual reality is still happily in Earth orbit). Note that in Orbiter, Mir is placed in an ecliptic orbit to make it a platform for interplanetary missions. This means that ISS and Mir have a very high relative inclination which makes the transfer very expensive in terms of fuel expenditure.

- ▶ Start Orbiter with the *Checklists|ISS to Mir* scenario. Your glider is docked to the ISS.
- ▶ Press **F1** to jump into the glider’s cockpit.
- ▶ Select target Mir in Orbit MFD: Press Right-**Shift T**, **Enter**, “Mir”.
- ▶ ISS and Mir orbits have a high relative inclination. To prepare for orbit change, select the Align plane mode in the left MFD (*SEL*, *Align planes*, and Left-**Shift T**, “Mir”).
- ▶ Undock from the ISS (**Ctrl D**). Once you are clear of the dock, close the nose cone (**K**).
- ▶ Switch to Orbit HUD mode (**H**).
- ▶ The first burn will take place at the DN (descending node) point. Use time compression to fast-forward there, but switch back to real-time when the “time-to-node” (Tn) value in the Align plane MFD is down to 500.
- ▶ Prepare attitude for the burn: click the “Orbit normal (+)” button. Your glider will now orient itself perpendicular to the orbital plane.

- ▶ When the “Engage engines” indicator in the Align MFD begins to flash, engage full main engines. The relative orbit inclination (RInc) should start to drop. Terminate the burn when the “Kill thrust” indicator appears and the inclination reaches its minimum.
- ▶ This is a very long burn (about 900 seconds), so you may want to fast-forward, but do not miss the end of the burn!
- ▶ You probably won’t be able to sufficiently reduce the inclination (less than 0.5°) in a single burn. Repeat the process at the AN (ascending node) point. Remember that the glider must be oriented in the opposite direction for this burn, by clicking the “Orbit Normal (-)” button.
- ▶ Once the orbital planes are aligned, you need to plot a rendezvous trajectory using the Sync Orbit MFD. The procedure is the same as in the previous mission.
- ▶ Tune your NAV1 receiver to MIR’s transponder frequency at 132.10, and NAV2 to the IDS frequency of Dock 1 at 135.00.
- ▶ Once the sync maneuver is complete, switch the HUD to Docking mode ( $\boxed{H}, \boxed{H}$ ), and switch one of the MFD displays to Docking (*SEL, Docking*). Slave both HUD and MFD to NAV1.
- ▶ Proceed with the docking maneuver to Mir in the same way as you did for docking at the ISS in the previous mission. Don’t forget to open the nose cone before making contact.

### 21.3 Mission 3: De-orbit from Mir

This mission completes your orbital roundtrip with a re-entry to return to Kennedy Space Center.

- ▶ Start Orbiter with the *Checklists/Deorbit* scenario. This picks up where the previous mission ended, with the glider docked to the Mir station. You are currently over the Pacific ocean, already at the correct location for the deorbit burn.
- ▶ Undock ( $\boxed{\text{Ctrl}} \boxed{D}$ ), and engage retros for a few seconds ( $\boxed{-} \boxed{Num}$ ) to get clear of the station.
- ▶ Close the nose cone ( $\boxed{K}$ ).
- ▶ Turn retrograde ( $\boxed{1}$ ).
- ▶ When the glider’s attitude has stabilised and the retrograde direction is no longer obstructed by the station, engage main engines at 100%.
- ▶ Kill engines when the perigee radius (PeR in Orbit MFD) has decreased to 5.600M.
- ▶ Turn prograde ( $\boxed{1}$ ).
- ▶ When attitude has stabilised, roll the glider level with the horizon ( $\boxed{L}$ ).
- ▶ Switch to Surface HUD mode ( $\boxed{H}$ ).
- ▶ Turn left MFD into Surface mode (*SEL, Surface*).

- ▶ You should reach 100 km altitude about 4000 km from the target (Dst: 4.000M in Map MFD). At this point, aerodynamic forces will become noticeable.
- ▶ At 50 km altitude, turn off attitude stabilisation (**L**), disable the RCS (**Ctrl** **I** **Num**), and make sure that “AF CTRL” is set to “ON”.
- ▶ Lift forces will cause the glider to pitch up. To bleed off energy you should perform left and right banks. Due to the relatively high lift/drag ratio of the glider you need very steep bank angles (90°).
- ▶ Your current flight path passes south of the KSC, so you should initially bank left to correct your approach path (check Map MFD).
- ▶ The bank angle will determine your rate of descent and airspeed. If you come up short to the KSC, reduce the bank angles to slow your descent and reduce atmospheric deceleration. If you come in too fast or too high, increase the bank angles to increase the descent slope and atmospheric friction.
- ▶ Timing of the reentry path is not quite as critical as for the Space Shuttle, because the glider can use its engines for a powered approach.
- ▶ When the distance to target drops below 500 km, tune your NAV1 receiver to frequency 112.70 (KSCX VOR), and NAV2 to frequency 134.20 (Rwy 33 ILS) using the COMMS mode (*SEL, COM/NAV*) in the right MFD.
- ▶ Turn the right MFD to Horizontal Situation Indicator (HSI) mode (*SEL, HSI*). Leave the left display slaved to NAV1, and flip the right display to NAV2 (Right-**Shift** **F**, Right-**Shift** **N**).
- ▶ Use the course deviation and glide slope indicators of the HSI displays for adjusting the approach path. They work like standard aircraft instruments.
- ▶ Lower landing gear (**G**). Deploy airbrakes (**Ctrl** **B**) as required. Touchdown speed is 150 m/s.
- ▶ Use wheel brakes (**.** and **.**) on rollout until you come to a halt.



*Rollout at the KSC SLF.*

## 22 Visual helpers

Orbiter has the ability to display a number of visual cues to provide additional data to the user. These include

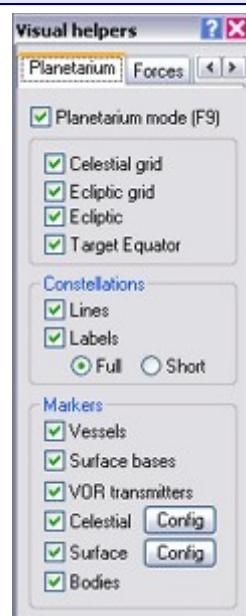
- ▶ a “Planetarium” mode that projects different coordinate grids onto the celestial sphere and provides markers and labels for various simulation objects, and celestial and planetary surface features.
- ▶ the display of force vectors on spacecraft
- ▶ the display of coordinate axes on different objects

The visual helper options can be configured via a dialog (**Ctrl F9**).

### 22.1 Planetarium mode

Lost in space? If you lose your bearings in the middle of an interplanetary flight, Orbiter offers guidance in the form of an in-flight “planetarium” with grids and object markers. To configure the planetarium options, open the *Visual helper* dialog (**Ctrl F9**) and select the *Planetarium* tab. A shortcut for turning the planetarium on and off is **F9**. The following items are available:

- ▶ Celestial grid lines (Earth equatorial reference frame)
- ▶ Ecliptic grid lines
- ▶ Ecliptic great circle
- ▶ Equator great circle of the target body (if applicable)
- ▶ Constellation lines and labels (full and abbreviated)
- ▶ Markers for celestial bodies
- ▶ Vessel markers
- ▶ Surface base markers
- ▶ Markers for navigation radio transmitter locations
- ▶ Markers for user-defined objects on the celestial sphere
- ▶ Markers for user-defined planetary surface labels



Some marker types will not be visible if the object is out of range, or from a planet surface during daylight.

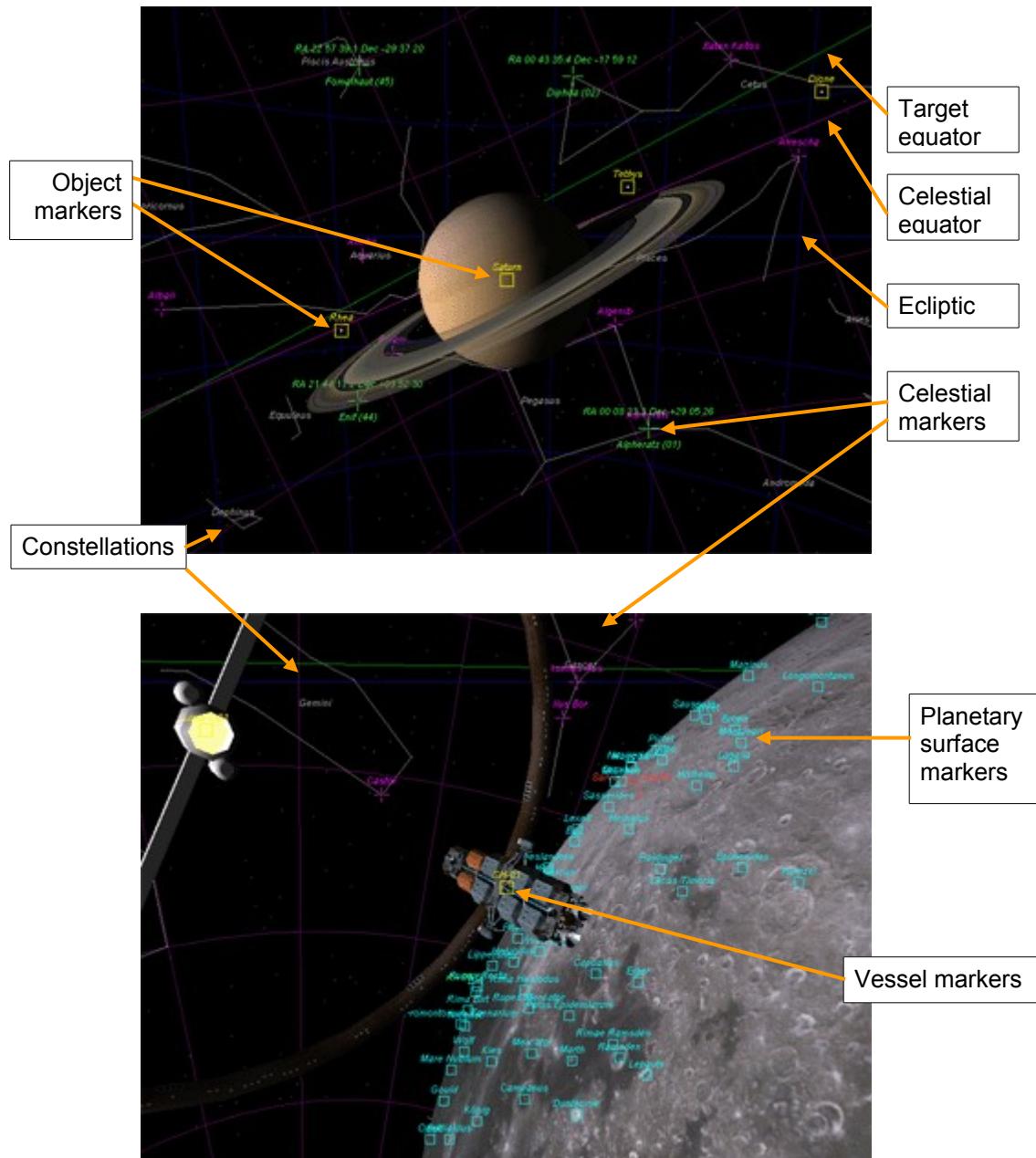
Planets may define their own sets of surface markers, to locate items such as natural landmarks, points of interest, historic landing sites, navigational aids, etc. Likewise, the planetary system may define sets of markers to identify bright stars, navigation stars, nebulae, etc. You can select marker sets by clicking the



"Config" button. This opens a further dialog box, where you can highlight the appropriate sets from a list box. Additional lists may be available as addons from Orbiter internet repositories. If you want to modify the provided marker sets or create your own, see [OrbiterConfig.pdf](#), Section "Adding custom markers".

Orbiter stores the current planetarium settings in its configuration file, and remembers them in the next simulation run.

Hardcore space simmers may spurn the planetarium as a cheat mode, but for the rest of us it can be a handy tool, and also helps in visualising the dynamics of planetary systems.



### *Grid lines, celestial, vessel and surface markers.*

## 22.2 Force vectors

Orbiter can provide a graphical display of the force vectors currently acting on a spacecraft. This option is particularly useful for educational applications, to provide a direct feedback of the effects of environmental parameters (gravity, atmosphere) and user input (e.g. change of lift as a function of changing angle of attack).

The display of force vectors can be activated and configured via the *Force* tab in the Visual helpers dialog ( ).

Tick *Body force vectors* to enable the vector display.

Orbiter allows to show a number of separate linear force components as well as the resulting total force:

Weight	G (yellow)	force due to gravitational field
Thrust	T (blue)	force generated by the vessel's propulsion system
Lift	L (green)	lift force generated by lifting airfoils in airflow
Drag	D (red)	drag force generated by motion through atmosphere
Total	F (white)	total force acting on the vessel

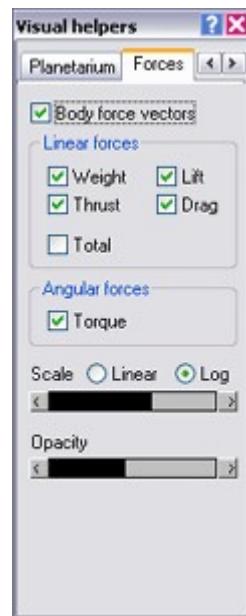
Note that the total force shown may not be equal to the sum of the four component forces, because additional forces may be acting on the vessel (e.g. user-defined forces).

Linear forces are shown graphically as vector arrows originating at the vessel's centre of gravity. The vector lengths are proportional to the force magnitudes, or logarithm of the magnitudes, depending on the *Scale* setting. The lengths can be adjusted with the provided slider. In addition, the magnitudes are also shown numerically in units of Newton [N].

In addition to the linear forces, Orbiter can also display the acting total torque,

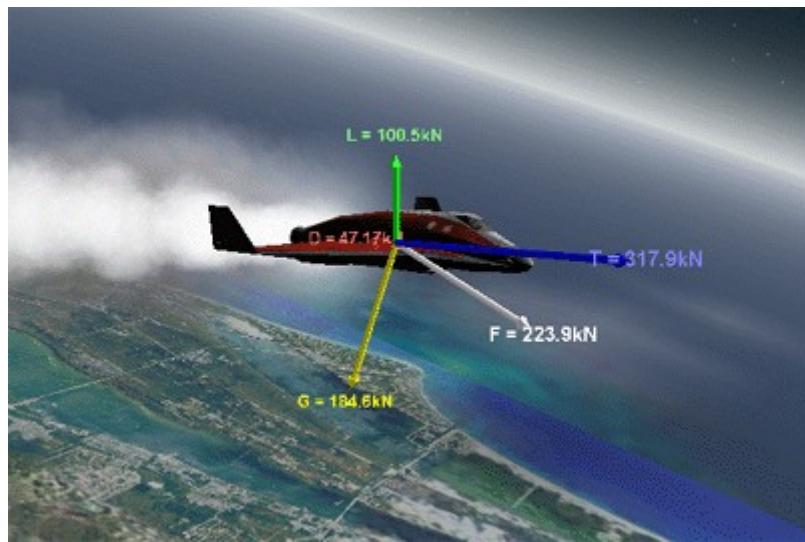
$$\mathbf{M} = \sum_i \mathbf{M}_i = \sum_i \mathbf{F}_i \times \mathbf{r}_i .$$

The torque vector is shown w.r.t. the centre of gravity of the vessel. The numerical value is shown in units of Newton × metre [Nm].



Note that for forces that are not generated at the vessel's centre of gravity (e.g. the lift vector), the total force displayed is broken up into a linear component originating at the centre of gravity, and a corresponding torque.

The opacity of the displayed force vectors can be adjusted with the *Opacity* slider, from completely transparent to completely opaque.



*Dynamics in action: A Delta-glider displaying the forces acting on its airframe.*

## 22.3 Coordinate axes

The orientation of the coordinate axes for the local frames of vessels, celestial bodies and spaceports can be displayed with the *Axes* tab of the Visual helpers dialog (**Ctrl F9**). Coordinate frames can be useful in particular for addon designers who want to make sure that the orientation of their spacecraft design within the simulator is correct.

The display of coordinate axes is enabled by ticking the *Coordinate axes* box.

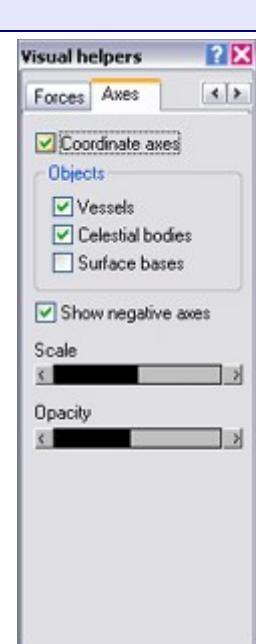
Axes can be displayed for

- ▶ vessels (spacecraft)
- ▶ celestial bodies (planets and moons)
- ▶ surface bases

Unless the *Show negative axes* box is ticked, only the positive x, y and z axes are displayed.

The length of the axis vectors can be adjusted with the *Scale* slider.

The opacity of the displayed vectors can be adjusted with the *Opacity* slider.



## 23 Demo mode

Orbiter can be run in "demo" or "kiosk" mode to facilitate its use in public environments such as exhibitions and museums.

Demo mode can be configured by manually editing the Orbiter.cfg configuration file in the main Orbiter directory. The following options are available:

Item	Type	Description
DemoMode	Bool	Set to TRUE to enable demo mode (default: FALSE)
BackgroundImage	Bool	Set to TRUE to cover the desktop with a static image. (default: FALSE)
BlockExit	Bool	Set to TRUE to disable the Exit function in Orbiter's launchpad dialog. If this option is enabled, Orbiter can only be exited via the task manager. (default: FALSE)
MaxDemoTime	Float	Defines the maximum runtime for a simulation (seconds). Orbiter automatically returns to the launchpad when the runtime has expired.
MaxLaunchpadIdleTime	Float	Maximum time spent in the launchpad without user input before Orbiter auto-launches a demo scenario (seconds)

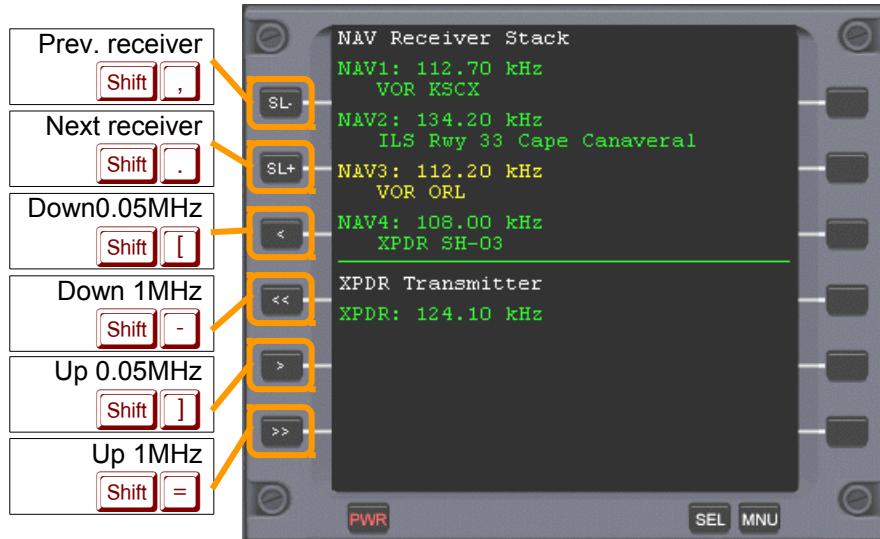
In demo mode, only the Scenario tab is accessible in the launchpad dialog, to prevent users from modifying simulation configuration features such as screen resolution or plugin modules. Orbiter should therefore be configured as required before launching into demo mode.

To use the auto-launch feature in demo mode, a folder "Demo" must be created in the main scenario folder (usually "Scenarios"). Orbiter will randomly pick a scenario from the Demo folder to launch.

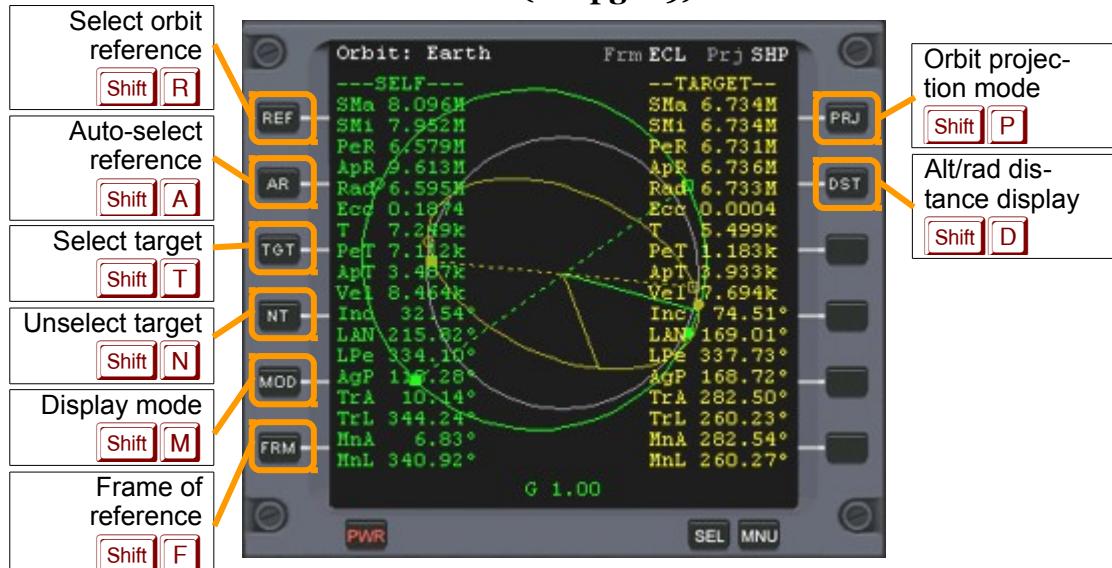
Note: When using Orbiter in kiosk mode, it is recommended to run the simulation in a window, or to use a fullscreen mode which matches the native PC screen resolution, to avoid excessive switching between video display modes.

## 24 MFD quick reference

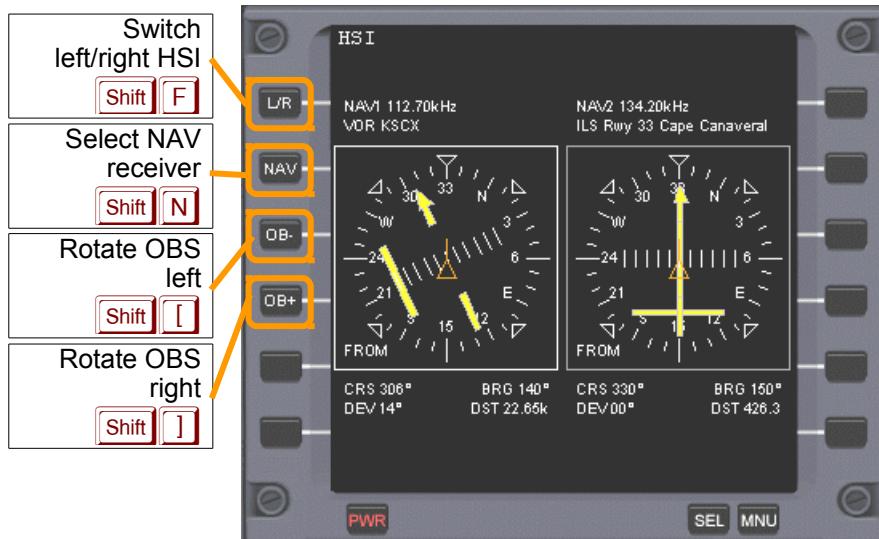
NAV/COM (see pg. 67)



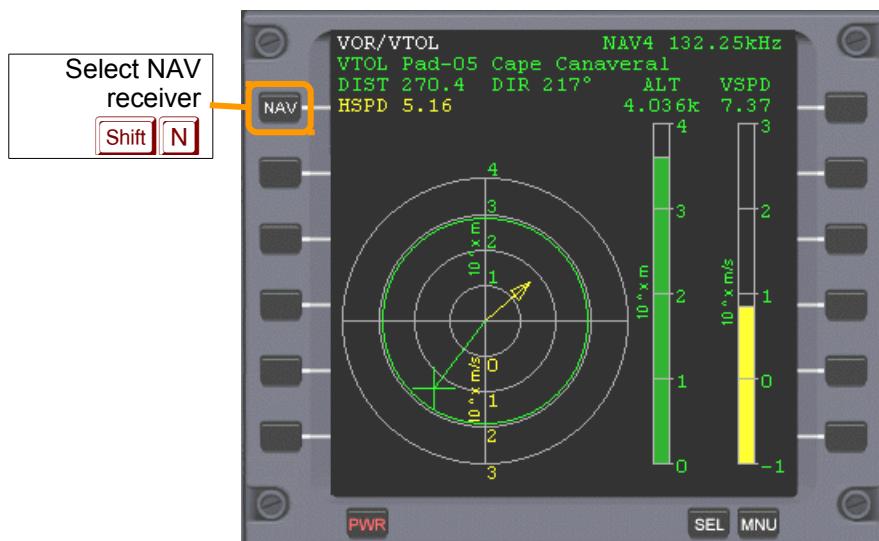
Orbit (see pg. 69)



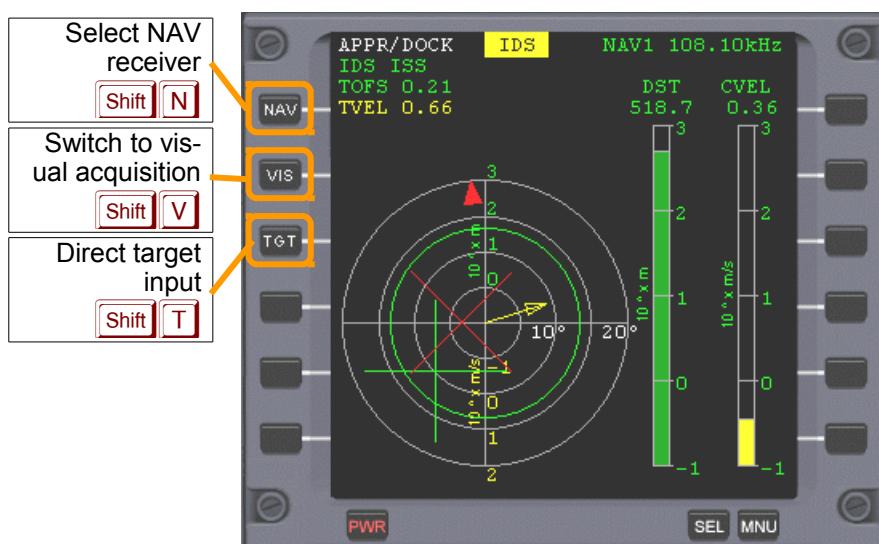
## HIS (see pg. 75)



## VOR/VTOL (see pg. 73)



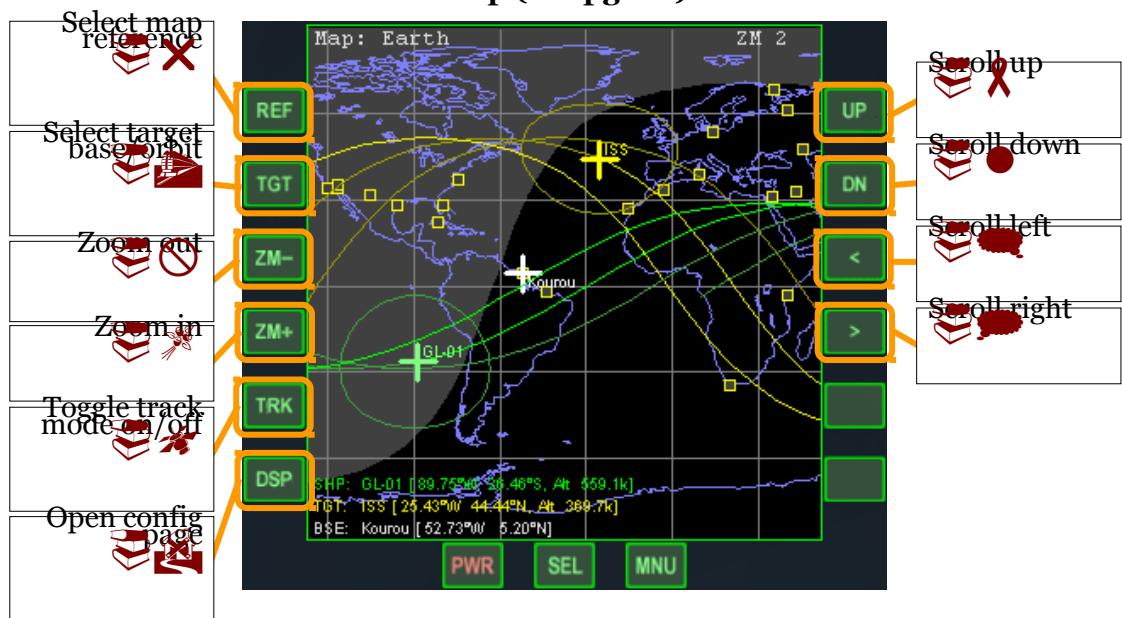
## Docking (see pg. 77)



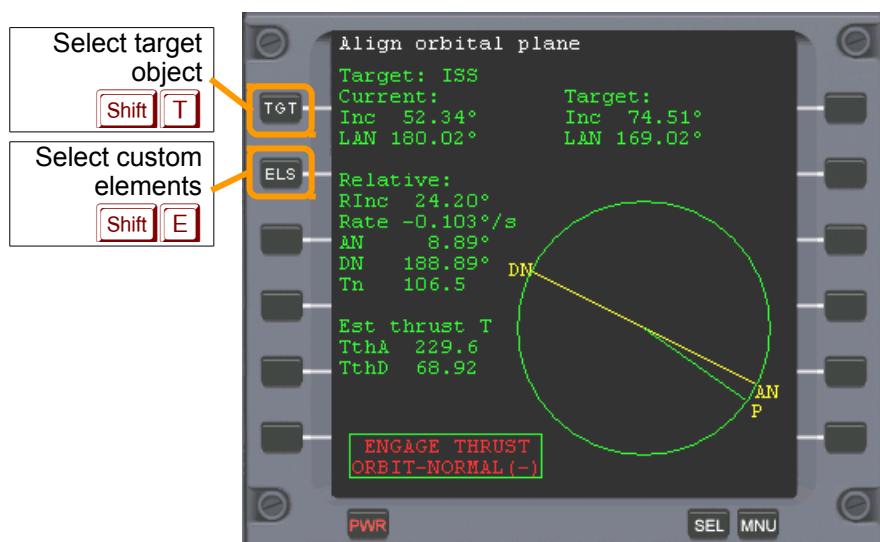
## Surface (see pg. 79)



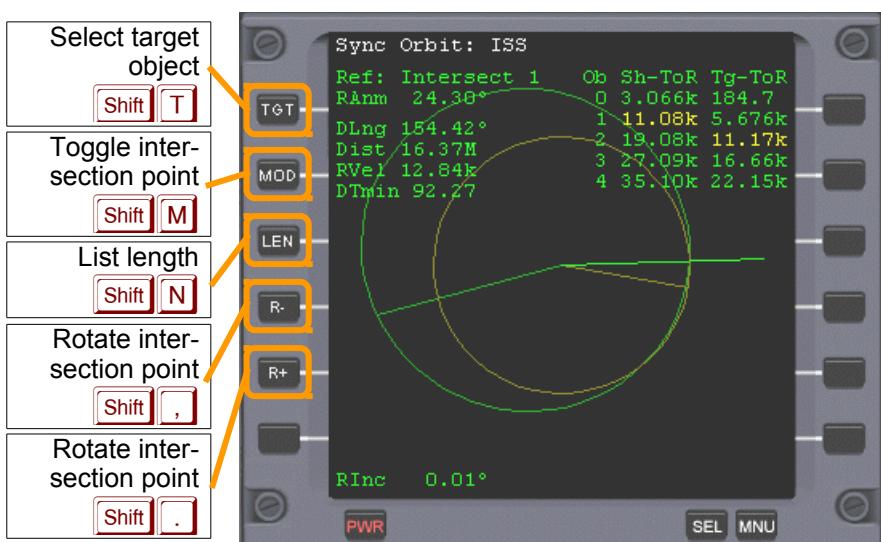
### Map (see pg. 82)



### Align orbital planes (see pg. 86)



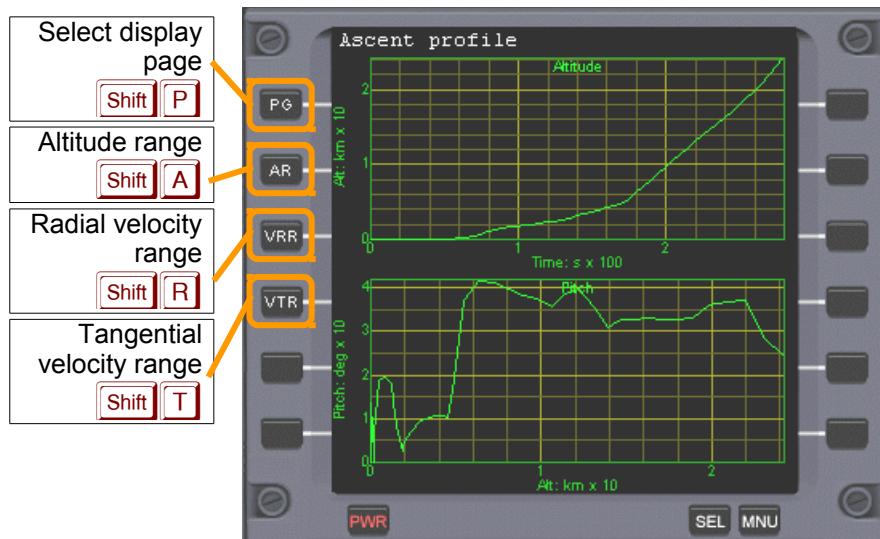
### Synchronise orbits (see pg. 87)



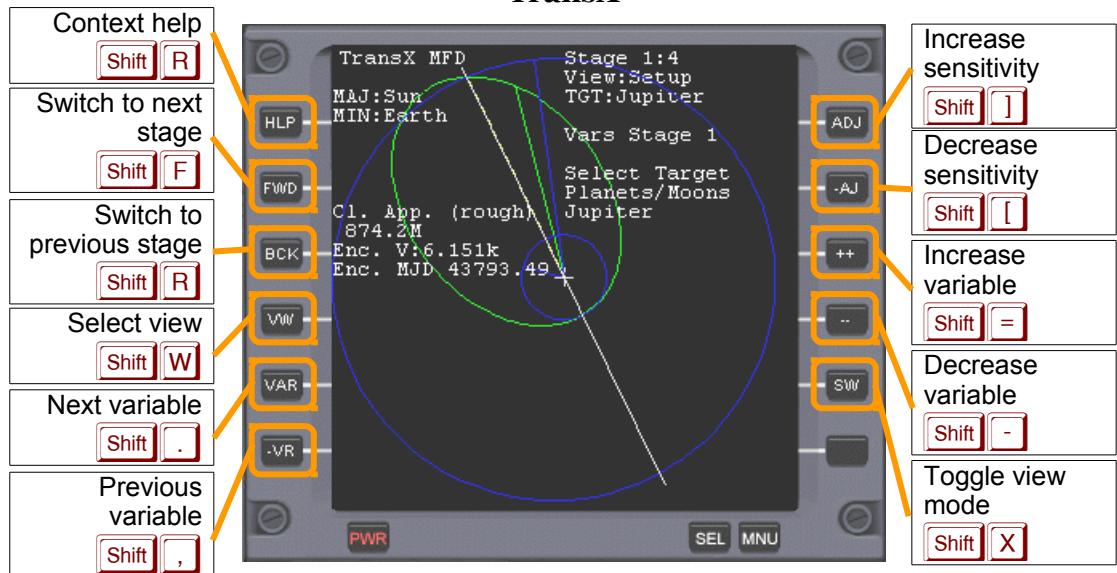
### **Transfer (see pg. 89)**



## Ascent



## TransX



## 25 Solar System: Constants and parameters

This section contains a list of physical and orbital planetary parameters used by Orbiter to build its solar system.

### 25.1 Astrodynamical constants and parameters

Constant	Symbol	Value
Julian day	d	86400 s
Julian year	yr	365.25 d
Julian century	Cy	36525 d
Speed of light	c	299792458 m/s
Gaussian gravitational constant	k	0.01720209895 $(\text{AU}^3/\text{d}^2)^{1/2}$

Table 1: Defining constants

Constant	Symbol	Value
Mean sidereal day		86164.09054 s = 23:56:04.09054
Sidereal year (quasar ref. frame)		365.25636 d
Light time for 1 AU	$\tau_A$	499.004783806 ( $\pm 0.00000001$ ) s
Gravitational constant	G	6.67259 ( $\pm 0.00030$ ) $\times 10^{-11}$ kg $^{-1}$ m $^3$ s $^{-2}$
General precession in longitude		5028.83 ( $\pm 0.04$ ) arcsec/Cy
Obliquity of ecliptic (J2000)	$\epsilon$	84381.412 ( $\pm 0.005$ ) arcsec
Mass: Sun / Mercury		6023600. ( $\pm 250.$ )
Mass: Sun / Venus		408523.71 ( $\pm 0.06$ )
Mass: Sun / (Earth+Moon)		328900.56 ( $\pm 0.02$ )
Mass: Sun / (Mars system)		3098708. ( $\pm 9.$ )
Mass: Sun / (Jupiter system)		1047.3486 ( $\pm 0.0008$ )
Mass: Sun / (Saturn system)		3497.898 ( $\pm 0.018$ )
Mass: Sun / (Uranus system)		22902.98 ( $\pm 0.03$ )
Mass: Sun / (Neptune system)		19412.24 ( $\pm 0.04$ )
Mass: Sun / (Pluto system)		1.35 ( $\pm 0.07$ ) $\times 10^8$
Mass: Moon / Earth		0.012300034 ( $\pm 3 \times 10^{-9}$ )

Table 2: Primary constants

Constant	Symbol	Value
Astronomical unit distance	$c \times \tau_A = \text{AU}$	$1.49597870691 \times 10^{11}$ ( $\pm 3$ ) m
Heliocentric gravitational constant	$k^2 \text{AU}^3 \text{d}^{-2} = GM_{\text{sun}}$	$1.32712440018 \times 10^{20}$ ( $\pm 8 \times 10^9$ ) m $^3$ s $^{-2}$
Mass: Earth / Moon		81.30059 ( $\pm 0.00001$ )

Table 3: Derived constants

#### Notes:

Data are from the 1994 IAU file of current best estimates. Planetary ranging determines the Earth/Moon mass ratio. The value for 1 AU is taken from JPL's current planetary ephemeris DE-405.

#### Reference:

Standish, E.M. (1995) ``Report of the IAU WGAS Sub-Group on Numerical Standards'', in Highlights of Astronomy (I. Appenzeller, ed.), Table 1, Kluwer Academic Publishers, Dordrecht.

### 25.2 Planetary mean orbits (J2000)

(Epoch = J2000 = 2000 January 1.5)

Planet (mean)	a [AU]	e	i [deg]	$\Omega$ [deg]	$\varpi$ [deg]	L [deg]
Mercury	0.38709893	0.20563069	7.00487	48.33167	77.45645	252.25084
Venus	0.72333199	0.00677323	3.39471	76.68069	131.53298	181.97973

<b>Earth</b>	1.000000011	0.01671022	0.00005	-11.26064	102.94719	100.46435
<b>Mars</b>	1.52366231	0.09341233	1.85061	49.57854	336.04084	355.45332
<b>Jupiter</b>	5.20336301	0.04839266	1.30530	100.55615	14.75385	34.40438
<b>Saturn</b>	9.53707032	0.05415060	2.48446	113.71504	92.43194	49.94432
<b>Uranus</b>	19.19126393	0.04716771	0.76986	74.22988	170.96424	313.23218
<b>Neptune</b>	30.06896348	0.00858587	1.76917	131.72169	44.97135	304.88003
<b>Pluto</b>	39.48168677	0.24880766	17.14175	110.30347	224.06676	238.92881

Table 4: Planetary mean orbits

### 25.3 Planetary orbital element centennial rates

(for the mean elements given above)

Planet (rate)	a [AU/Cy]	e [1/Cy]	i ["/Cy]	$\Omega$ ["/Cy]	$\varpi$ ["/Cy]	L ["/Cy]
<b>Mercury</b>	0.00000066	0.00002527	-23.51	-446.30	573.57	538101628.29
<b>Venus</b>	0.00000092	-0.00004938	-2.86	-996.89	-108.80	210664136.06
<b>Earth</b>	-0.00000005	-0.00003804	-46.94	-18228.25	1198.28	129597740.63
<b>Mars</b>	-0.00007221	0.00011902	-25.47	-1020.19	1560.78	68905103.78
<b>Jupiter</b>	0.00060737	-0.00012880	-4.15	1217.17	839.93	10925078.35
<b>Saturn</b>	-0.00301530	-0.00036762	6.11	-1591.05	-1948.89	4401052.95
<b>Uranus</b>	0.00152025	-0.00019150	-2.09	-1681.40	1312.56	1542547.79
<b>Neptune</b>	-0.00125196	0.0000251	-3.64	-151.25	-844.43	786449.21
<b>Pluto</b>	-0.00076912	0.00006465	11.07	-37.33	-132.25	522747.90

" arcsecs

Cy Julian century

a Semi-major axis

e eccentricity

i inclination

$\Omega$  longitude of the ascending node

$\varpi$  longitude of perihelion

L mean longitude

#### Notes:

This table contains mean orbit solutions from a 250 yr. least squares fit of the DE 200 planetary ephemeris to a Keplerian orbit where each element is allowed to vary linearly with time. This solution fits the terrestrial planet orbits to ~25" or better, but achieves only ~600" for Saturn. Elements are referenced to mean ecliptic and equinox of J2000 at the J2000 epoch (2451545.0 JD).

#### Reference:

Explanatory Supplement to the Astronomical Almanac. 1992. K. P. Seidelmann, Ed., p.316 (Table 5.8.1), University Science Books, Mill Valley, California.

### 25.4 Planets: Selected physical parameters

Planet	Mean radius [km]	Mass [ $10^{23}$ kg]	Density [g/cm <sup>3</sup> ]	Siderial rotation period [h]	Siderial orbit period [yr]
<b>Mercury</b>	2440. ±1.	3.301880	5.427	1407.509	0.2408445
<b>Venus</b>	6051.84 ±0.01	48.6855374	5.204	-5832.444	0.6151826
<b>Earth</b>	6371.01 ±0.02	59.73698968	5.515	23.93419**	0.9999786
<b>Mars</b>	3389.92 ±0.04	6.418542	3.9335±0.0004	24.622962	1.88071105
<b>Jupiter</b>	69911. ±6.	18986.111	1.326	9.92425	11.856523
<b>Saturn</b>	58232. ±6.	5684.6272	0.6873	10.65622	29.423519
<b>Uranus</b>	25362. ±12.	868.32054	1.318	17.24 ±0.01	83.747407
<b>Neptune</b>	24624. ±21.	1024.569	1.638	16.11 ±0.01	163.72321
<b>Pluto*</b>	1151	0.15	1.1	153.28	248.0208

Planet	V(1,0) [mag.]	Geometric albedo	Equatorial gravity [m/s <sup>2</sup> ]	Escape velocity [km/s]
<b>Mercury</b>	-0.42	0.106	3.701	4.435
<b>Venus</b>	-4.4	0.65	8.87	10.361
<b>Earth</b>	-3.86	0.367	9.780327	11.186
<b>Mars</b>	-1.52	0.15	3.69	5.027
<b>Jupiter</b>	-9.4	0.52	23.12 ± 0.01	59.5
<b>Saturn</b>	-8.88	0.47	8.96 ± 0.01	35.5
<b>Uranus</b>	-7.19	0.51	8.69 ± 0.01	21.3
<b>Neptune</b>	-6.87	0.41	11.00 ± 0.05	23.5
<b>Pluto</b>	-1.0*	0.3*	0.655	1.3

All values from reference [1] except Pluto data from [2]. Mercury to Neptune masses derived from GM data in [1] (thanks to Duncan Sharpe for pointing this out).

\*\* Orbiter now uses 23.93447h (= 23h 56m 4.09s) which appears to give better long term stability.

### References

[1] Yoder, C.F. 1995. ``Astrometric and Geodetic Properties of Earth and the Solar System'' in Global Earth Physics, A Handbook of Physical Constants, AGU Reference Shelf 1, American Geophysical Union.

[2] Explanatory Supplement to the Astronomical Almanac. 1992. K. P. Seidelmann, Ed., p.706 (Table 15.8), University Science Books, Mill Valley, California.

## 25.5 Rotation elements

Planet	North pole		Obliquity of ecliptic* [°]	Longitude of Sun's transit* [°]
	Right ascension $\alpha_*$ [°]	Declination $\delta_*$ [°]		
<b>Mercury</b>	280.99	61.44	7.01	228.31
<b>Venus</b>	272.78	67.21	1.27	302.07
<b>Earth</b>	-	90	23.44	0
<b>Mars</b>	317.61	52.85	26.72	262.78
<b>Jupiter</b>	268.04	64.49	2.22	157.68
<b>Saturn</b>	40.14	83.50	28.05	349.39
<b>Uranus</b>	257.29	-15.09	82.19	167.62
<b>Neptune</b>	295.25	40.63	29.48	221.13
<b>Pluto</b>	311.50	4.14	68.69	225.19

Reference:

The Astronomical Almanac 1990 (North pole coordinates)

(\*) Derived from north pole coordinates (MS)

## 25.6 Atmospheric parameters

Planet	Surface pressure [kPa]	Surface density [kg/m <sup>3</sup> ]	Scale height [km]	Avg. temp [K]	Wind speeds [m/s]
<b>Mercury</b>					
<b>Venus</b>	9200	~65	15.9	737	0-3-1 (surface)
<b>Earth</b>	101.4	1.217	8.5	288	0-100
<b>Mars</b>	0.61 (variable)	~0.020	11.1	~210	0-30
<b>Jupiter</b>	>> 10 <sup>4</sup>	~0.16 at 1 bar	27	~129 ~165 at 1 bar	up to 150 at < 30° latitude up to 40 else
<b>Saturn</b>	>> 10 <sup>4</sup>	~0.19 at 1 bar	59.5	~97 ~134 at 1 bar	up to 400 at < 30° latitude up to 150 else
<b>Uranus</b>	>> 10 <sup>4</sup>	~0.42 at 1 bar	27.7	~58	0-200

				$\sim 76$ at 1 bar	
<b>Neptune</b>	$>> 10^4$	$\sim 0.45$ at 1 bar	19.1-20.3	$\sim 58$ $\sim 72$ at 1 bar	0-200
<b>Pluto</b>					

## 26 Calculation of orbital elements

Six scalar parameters (“elements”) are required to define the shape of an elliptic orbit, its orientation in space and a location along its trajectory.

- a      Semi-major axis
- e      Eccentricity
- i      Inclination
- $\Omega$     Longitude of ascending node
- $\omega$     argument of periapsis
- $\nu$     true anomaly

### 26.1 Calculating elements from state vectors

Let  $\mathbf{r}$  and  $\mathbf{v}$  be the cartesian position and velocity vectors of an orbiting object in coordinates of a reference frame with respect to which the elements of the orbit are to be calculated (e.g. geocentric equatorial for an orbit around Earth, or heliocentric ecliptic for an orbit around the Sun). We assume a right-handed system with the x-axis pointing towards the vernal equinox (or other reference direction) and the z-axis pointing upwards.

Compute the following auxiliary vectors:

$$\mathbf{h} = \mathbf{r} \times \mathbf{v} = (r_y v_z - r_z v_y, -r_x v_z + r_z v_x, r_x v_y - r_y v_x)$$

$$\mathbf{n} = \mathbf{z} \times \mathbf{h} = (-h_y, h_x, 0)$$

$$\mathbf{e} = \frac{1}{\mu} \left[ \left( v^2 - \frac{\mu}{|\mathbf{r}|} \right) \mathbf{r} - (\mathbf{r} \cdot \mathbf{v}) \mathbf{v} \right]$$

where  $\mathbf{h}$  is a vector perpendicular to the orbital plane,  $\mathbf{n}$  points towards the ascending node (the z-component of  $\mathbf{n}$  is zero), and  $\mathbf{e}$  is the eccentricity vector (pointing towards the periapsis) with  $\mu = GM$ ,  $G$  is the gravitational constant and  $M$  is the mass of the central body (neglecting the mass of the orbiter).

#### Semi-major axis:

$$a = \frac{-\mu}{2E} \text{ with } E = \frac{v^2}{2} - \frac{\mu}{|\mathbf{r}|}$$

#### Eccentricity:

$$e = |\mathbf{e}| \text{ or } e = \sqrt{1 + \frac{2Eh^2}{\mu^2}}$$

#### Inclination:

$$i = \arccos \frac{h_z}{|\mathbf{h}|}$$

#### Longitude of ascending node:

$$\Omega = \arccos \frac{n_x}{|\mathbf{n}|} \quad (\text{if } n_y < 0 \text{ then } \Omega = 2\pi - \Omega)$$

$\Omega$  is the angle between reference direction (1,0,0) (e.g. vernal equinox) and the ascending node.

$\Omega$  is undefined for equatorial orbits ( $i = 0$ ), in which case Orbiter by convention sets  $\Omega = 0$ , i.e. it places the ascending node in the reference direction, which is equivalent to setting  $\mathbf{n}/|\mathbf{n}| = (1,0,0)$ .

### Argument of periapsis:

$$\omega = \arccos \frac{\mathbf{e} \cdot \mathbf{r}}{|\mathbf{e}| |\mathbf{r}|} \quad (\text{if } e_z < 0 \text{ then } \omega = 2\pi - \omega)$$

$\omega$  is the angle between the ascending node and the periapsis.  $\omega$  is undefined for equatorial orbits in which case according to above convention we get

$$\omega = \arccos \frac{e_x}{|\mathbf{e}|} \quad (\text{if } e_z < 0 \text{ then } \omega = 2\pi - \omega)$$

$\omega$  is also undefined for circular orbits in which case Orbiter by convention places the periapsis at the ascending node, i.e.  $\omega = 0$ .

### True anomaly:

$$v = \arccos \frac{\mathbf{e} \cdot \mathbf{r}}{|\mathbf{e}| |\mathbf{r}|} \quad (\text{if } \mathbf{r} \cdot \mathbf{v} < 0 \text{ then } v = 2\pi - v)$$

$v$  is the angle between the periapsis and object position. Note that this expression is undefined for circular orbits, in which case the periapsis coincides with the ascending node according to the convention above, i.e.

$$v = \arccos \frac{\mathbf{n} \cdot \mathbf{r}}{|\mathbf{n}| |\mathbf{r}|} \quad (\text{if } \mathbf{n} \cdot \mathbf{r} > 0 \text{ then } v = 2\pi - v)$$

If in addition the inclination is zero then the true anomaly further simplifies to

$$v = \arccos \frac{r_x}{|\mathbf{r}|} \quad (\text{if } r_x > 0 \text{ then } v = 2\pi - v)$$

Some dependent parameters can be derived from the above elements:

### Linear eccentricity:

$$\varepsilon = a e$$

### Semi-minor axis:

$$b^2 = a^2(1 - e^2)$$

### Periapsis and apoapsis distances:

$$d_p = a(1 - e)$$

$$d_a = a(1 + e)$$

### Longitude of the periapsis:

$$\varpi = \Omega + \omega$$

**Eccentric anomaly:**

$$E = \arccos \frac{1 - |\mathbf{r}|/a}{e}$$

**Mean anomaly:**

$$M = E - e \sin E$$

**Mean longitude:**

$$L = M + \varpi$$

**True longitude:**

$$l = \varpi + v$$

**Orbit period:**

$$T = 2\pi \sqrt{a^3/\mu}$$

## **27 Terms of Use**

The ORBITER software, documentation and the content on the ORBITER website is copyright 2000-2010 by Martin Schweiger.

ORBITER is not in the public domain: it is the intellectual property of Martin Schweiger.

### **27.1 Orbiter Freeware License**

The licensor of Orbiter grants you a limited non-exclusive license to use, copy and distribute Orbiter, downloadable from [orbit.medphys.ucl.ac.uk](http://orbit.medphys.ucl.ac.uk), without fee for personal, educational, charity and other non-commercial use.

You agree to comply with the following conditions:

- ▶ You will not modify any of the binary codes of the Orbiter software in any way. If you publish a library module for Orbiter that uses undocumented features of the orbiter programming interface, or links to the orbiter core in an undocumented way, you must state this in the documentation for the module, and advise users of potential incompatibility of the module with future Orbiter versions.
- ▶ You will not sell Orbiter or any parts of it, or charge others for use of it (either for profit or merely to recover your media and distribution costs) whether as a stand-alone product, as part of a compilation or anthology, or included in a separate software package, without explicit prior written permission by the licensor.
- ▶ You will not remove or alter any copyright or license notices contained in the software and documentation, or remove or alter any identifying elements, including splash screens and logos, or try to hide or reassign in any way the name or origin of the software or any of its components.
- ▶ You will not use Orbiter or any parts of it to advertise, promote, present or sell any software or other product or service without explicit prior written permission by the licensor.
- ▶ You will not use Orbiter to engage in or allow others to engage in any illegal activity.

### **27.2 Disclaimer of warranty**

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS 'AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CON-

TRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.