

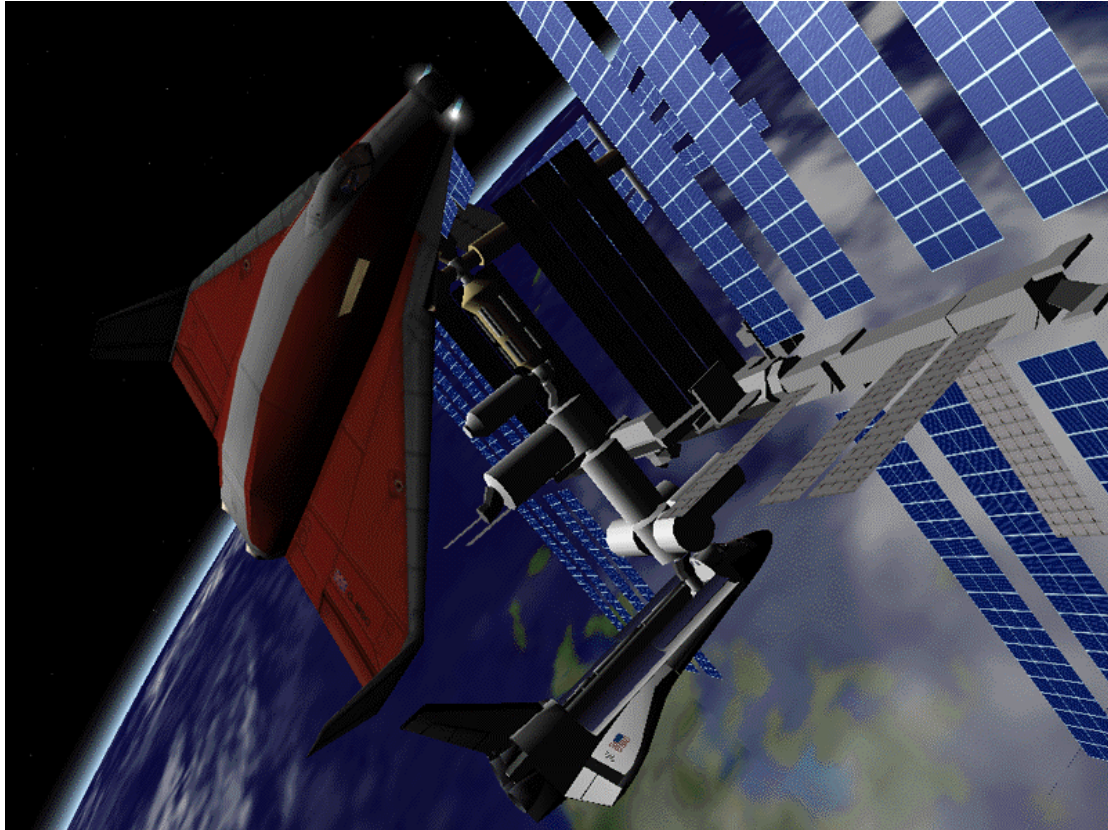
ORBITER User Manual



©2000-2003 Martin Schweiger

www.medphys.ucl.ac.uk/~martins/orbit/orbit.html

06 November 2003



Copyright

The ORBITER software, documentation and the content on the ORBITER website is copyright 2000-2003 by Martin Schweiger.

ORBITER is free software in the sense that you are free to download, copy and redistribute it for personal and non-commercial purposes, provided that the copyright notice is retained in the copy. You are not allowed to charge a fee for the software without the consent of the author other than that to cover the cost of the distribution. If a fee is charged it must be made clear to the purchaser that the software is freeware and that the fee is to cover the distributor's costs of providing the software. Selling ORBITER or parts of it or bundling it into a commercial product or using ORBITER to promote a commercial product without the author's consent is an infringement of the author's copyrights.

You are not allowed to modify the Orbiter binary code or the documentation distributed with the Orbiter software package. You are however allowed to distribute modifications to configuration scripts, meshes, or the sample source codes in the `Orbitersdk\samples` folder, provided that you make it clear to the recipients that you have made such modifications.

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

Disclaimer

The ORBITER software is provided as is without any warranty of any kind. To the maximum extent permitted by applicable law, Martin Schweiger further disclaims all warranties, including without limitation any implied or stated warranties of merchantability, fitness for a particular purpose, and noninfringement. The entire risk arising out of the use or performance of this product and documentation remains with recipient. To the maximum extent permitted by applicable law, in no event shall program ORBITER or its suppliers be liable for any consequential, incidental, direct, indirect, special, punitive, recursive, or other damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, personal injury, disruption of family life, or other pecuniary loss) arising out of this agreement or the use of or inability to use the product, even if program ORBITER has been advised of the possibility of such damages. Because some states/jurisdictions do not allow the exclusion or limitation of liability of consequential or incidental damages, the above limitation may not apply to the recipient.

Contents

1	INTRODUCTION	5
2	INSTALLATION.....	6
2.1	Hardware requirements	6
2.2	Download	6
2.3	Installation	6
2.4	Uninstall	7
3	THE LAUNCHPAD	8
3.1	Scenario tab.....	8
3.2	Parameters tab	9
3.3	Visual effects tab.....	10
3.4	Modules tab	11
3.5	Video tab.....	12
3.6	Joystick tab	13
4	QUICKSTART	14
5	KEYBOARD INTERFACE.....	19
5.1	General	19
5.2	Spacecraft controls	20
5.3	External views.....	21
5.4	Internal (cockpit) view	21
5.5	Menu selections	23
6	JOYSTICK INTERFACE	24
7	MOUSE INTERFACE	25
8	SPACECRAFT CLASSES	26
8.1	Delta-glider.....	26
8.2	Shuttle-A	28
8.3	Shuttle PB (PTV).....	29
8.4	Dragonfly.....	29
8.5	Space Shuttle Atlantis.....	30
8.6	International Space Station (ISS)	33
8.7	Space Station MIR	33
8.8	Lunar Wheel Station	34
8.9	Hubble Space Telescope.....	35
8.10	LDEF Satellite	36
9	CAMERA MODES	37
9.1	Internal view.....	37
9.2	External views	37
9.3	Selecting the field of view	38
9.4	Storing and recalling camera modes	39
10	HEAD-UP DISPLAY	41
10.1	General information display	41
10.2	Engine information display.....	42
10.3	Surface mode.....	42
10.4	Orbit mode	42
10.5	Docking mode	42
11	MULTIFUNCTIONAL DISPLAY MODES	44
11.1	COM/NAV receiver setup	44
11.2	Orbit	45
11.3	VOR/VTOL.....	48
11.4	Horizontal Situation Indicator	49

11.5	Docking	50
11.6	Surface.....	52
11.7	Map	54
11.8	Align orbital plane	55
11.9	Synchronise orbit	55
11.10	Transfer.....	56
11.11	Ascent profile (custom MFD mode)	59
12	SPACECRAFT CONTROLS	60
12.1	Main, retro and hover engines	60
12.2	Attitude thrusters.....	61
13	RADIO NAVIGATION AIDS	62
14	BASIC FLIGHT MANOEUVRES.....	63
14.1	Surface flight	63
14.2	Launching into orbit.....	63
14.3	Changing the orbit.....	64
14.4	Rotating the orbital plane.....	65
14.5	Synchronising orbits	66
14.6	Landing (runway approach)	67
14.7	Docking	68
15	EXTRA FUNCTIONALITY.....	70
15.1	Frame rate monitor	70
15.2	Remote vessel control	71
15.3	Flight data monitor	71
16	FLIGHT CHECKLISTS	73
16.1	Mission 1: Delta-glider to ISS	73
16.2	Mission 2: ISS to MIR transfer	75
16.3	Mission 3: De-orbit from MIR	76
17	ORBITER CONFIGURATION	77
17.1	Master configuration file.....	77
17.2	Planetary systems	78
17.3	Planets	78
17.4	Surface bases	81
17.5	Adding objects to surface bases.....	82
17.6	Scenario files	86
17.7	Migrating from station to vessel definitions.....	90
18	PLANET SURFACE TEXTURES.....	92
18.1	Texture format	92
18.2	Where ORBITER looks for textures.....	93
18.3	Using <i>ptex</i> to generate custom planet textures	93
APPENDIX A	SOLAR SYSTEM: CONSTANTS AND PARAMETERS	94
A.1	Astrodynamic constants and parameters	94
A.2	Planetary mean orbits (J2000).....	94
A.3	Planetary orbital element centennial rates	95
A.4	Planets: Selected physical parameters	95
A.5	Rotation elements.....	96
A.6	Atmospheric parameters.....	96
APPENDIX B	CALCULATION OF ORBITAL ELEMENTS.....	97
B.1	Calculating elements from state vectors.....	97

1 Introduction

ORBITER is a free flight simulator that goes beyond the confines of Earth's atmosphere. Launch the Space Shuttle from Kennedy Space Center to deploy a satellite, rendezvous with the International Space Station or that the futuristic Delta-glider for a tour through the solar system – the choice is yours.

But make no mistake – ORBITER is not a space shooter. The emphasis is firmly on realism and the learning curve can be steep. Be prepared to invest some time and effort to brush up on your orbital mechanics background, and dig your calculator.

At the very least, you should familiarise yourself with the primary spacecraft controls and instrumentation in this manual to get your ship off the ground. "Advanced" missions like rendezvous manoeuvres or interplanetary trips will require a lot more effort!

Suggestions, corrections, bug reports (and praise) for the ORBITER software or this documentation are always welcome. The preferred way to post your comments, in particular if they may be of interest to other users, is via the ORBITER mailing list or the ORBITER web forum, available via links from the official Orbiter site

www.medphys.ucl.ac.uk/~martins/orbit/orbit.html (alias www.orbitersim.com). Unfortunately I can't guarantee to answer all Orbiter-related mail sent directly to me.

Before posting a bug report make sure you have got the latest ORBITER release, and that your problem is not already documented in the FAQ, the bug tracker or the bug forum (all accessible from the ORBITER home page).

ORBITER – the thinking being's simulator.

Enjoy the ride!

Martin Schweiger

2 Installation

2.1 Hardware requirements

The standard ORBITER distribution requires the following minimum hardware features:

- 300MHz PC or better (Pentium, Athlon, etc.)
- 128MB RAM or more
- Windows 95/98/Me/2000/XP
- DirectX 7.0 or higher
- DirectX compatible 3D graphics accelerator card with at least 16MB of video RAM (32MB or more recommended) and DXT texture compression support
- Approximately 60MB of free disk space for the minimum installation (additional high-resolution textures and addons will require more space).
- DirectX compatible joystick (optional)

Installing high-resolution texture packs or addons may have an impact on performance and could require significantly higher computing specs.



Since ORBITER keeps evolving these specs tend to become obsolete over time. If you don't get a reasonable frame rate (say ≥ 20 fps) using the default Orbiter.cfg on a machine which meets the specs then please drop me a note and I will correct the requirement list upward.

2.2 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://www.medphys.ucl.ac.uk/~martins/orbit/orbit.html>. Orbiter is distributed in several packages (.zip files). The *Base* and *Textures* packages are required, other packages are optional.

All package names contain a 6-digit time stamp (YYMMDD) which allows to identify the modification date of the package. For example, orbiter031020_base.zip contains the base package built on October 20, 2003. 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.

2.3 Installation

- Create a new folder for the ORBITER installation, e.g. \Program Files\Orbiter_031020.
- 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.
- Download the *Base* package and *Textures* packages from an ORBITER download site into your new ORBITER folder and unzip them with WinZip or an equivalent utility.
Important: Take care to preserve the directory structure of the packages (for example, in WinZip this requires to activate the "Use Folder Names" option).
- After unzipping the packages, 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 "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.

2.4 Uninstall

- ORBITER does not modify the Windows registry or any system resources, so no complicated uninstallation process is required. Simply delete the Orbiter folder with all contents and subdirectories. This will uninstall ORBITER completely.

3 The Launchpad

Starting Orbiter.exe brings up the *Orbiter Launchpad* dialog box. From here, you can

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

From the Launchpad dialog, the simulation can be started by pressing the “ORBITER” button, provided a scenario has been selected. If you are running Orbiter for the first time, you should make sure that the simulation parameters (in particular video options) are properly set before starting the simulation.

3.1 Scenario tab

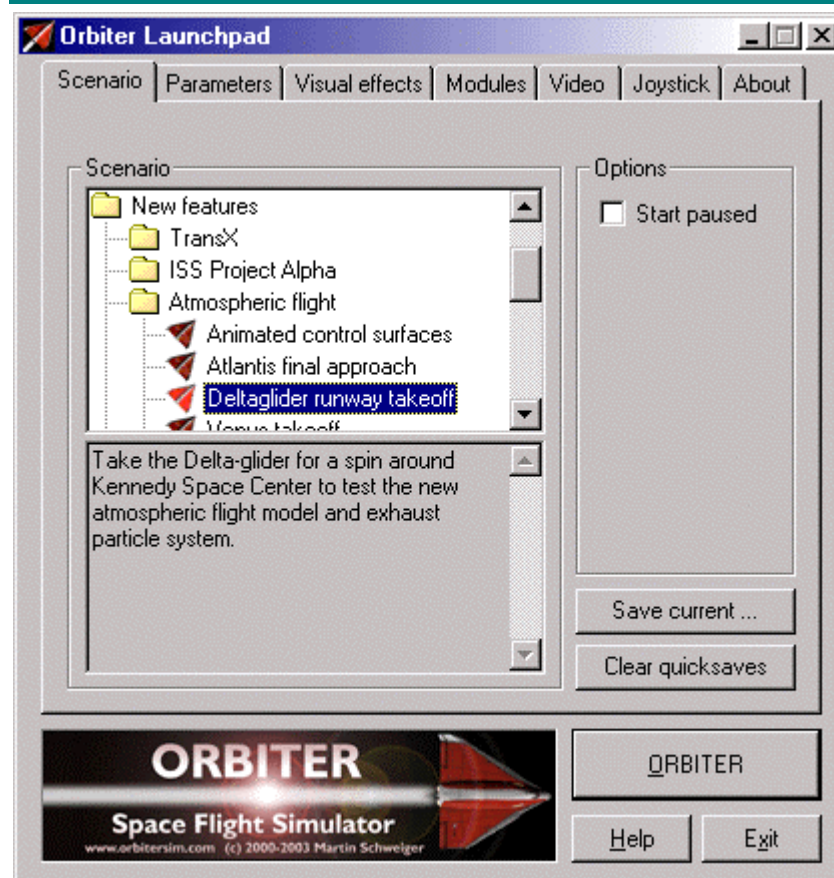


Figure 1: Launchpad dialog, Scenario tab

Scenario:

Contains a hierarchical list of available scenarios. Select one and launch it with the “Orbiter” button. Below the list is a description of the currently selected scenario.

Special scenarios:

- The **(Current state)** scenario is automatically generated whenever you exit the simulator. Use this to continue from the latest exit state.
- 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!

NEW!

Options:

- **Start paused:** Pause simulation on start. Press **Ctrl** **P** to unpause after launch.

Save current:

Save the current exit state under a new name with a custom description.

Clear quicksaves:

Empty the Quicksave folder.

3.2 Parameters tab

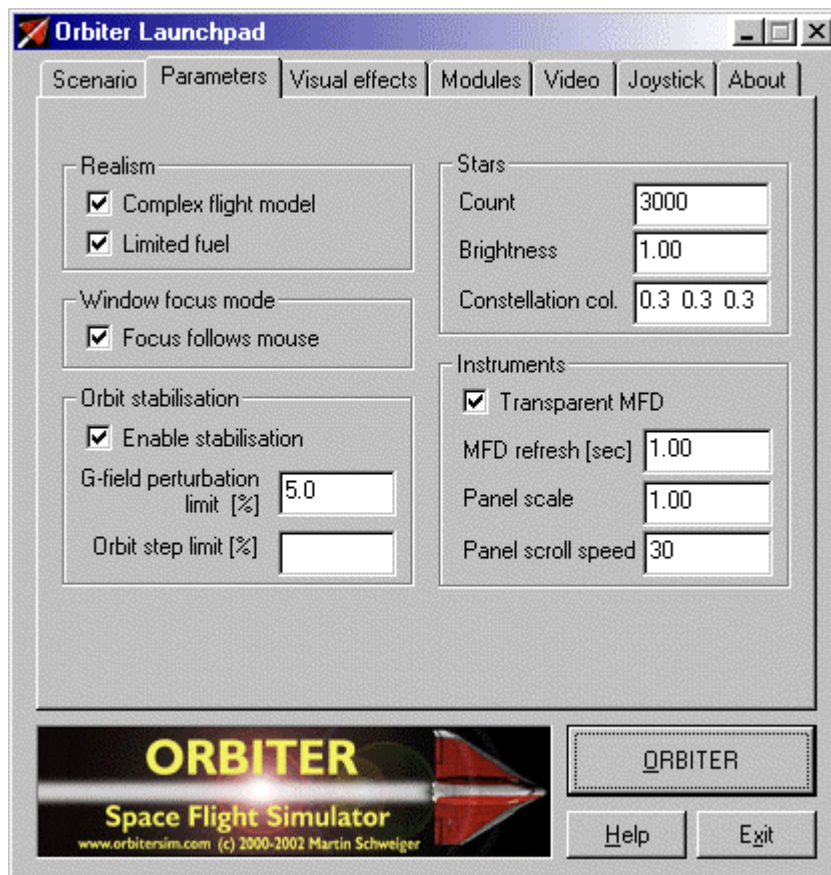


Figure 2: Launchpad dialog, Parameters tab

Realism

- **Complex flight model:** Select the realism of the flight model for spacecraft.
- **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.

NEW!

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.

NEW!

Orbit stabilisation

- **Enable stabilisation:** If this option is enabled, Orbiter uses a simplified analytic 2-body method to update the state vectors of orbiting bodies under certain conditions. This can help to avoid orbit deterioration at high time compressions.

- **G-field perturbation limit:** Defines the upper limit of perturbation [%] of the gravity field of the main gravity source under which stabilisation is enabled. A higher value will switch to stabilisation mode even if the 2-body assumption is not very accurate.
- **Orbit step limit:** The “orbit step” is the fraction [%] of a full orbit an object advances in a single frame. If a step is larger than the specified limit, orbit stabilisation is activated.

Stars

- **Count:** Number of displayed background stars. Orbiter contains a database of about 16000 bright stars. Specifying a large number will provide a more impressive night sky, but may degrade performance. Set to zero to suppress background stars.
- **Brightness:** Brightness scaling factor for background stars. Valid range is 0.2 - 2. Note that the dynamic range becomes less realistic for large values.
- **Constellation col:** Colour (RGB triplet) to use for drawing constellations (F9). Valid range for each value is 0 - 1.

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.
- **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.

3.3 Visual effects tab

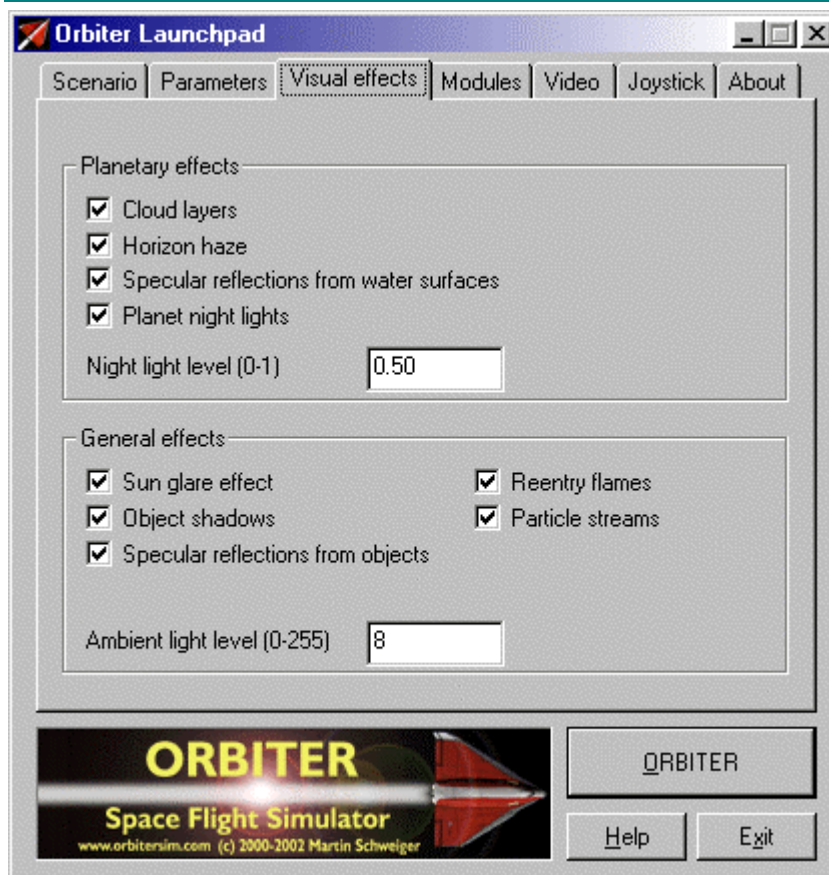


Figure 3: Launchpad dialog, Visual effects tab

Planetary effects

- **Cloud layers:** Render clouds as a separate mesh layer for appropriate planets. May degrade performance.
- **Horizon haze:** Render intensity-graded (“glowing”) horizon layer for planets with atmospheres. May degrade performance.
- **Specular reflections from water surfaces:** Render water surfaces on planets with specular reflection effects. May degrade performance.
- **Planet night lights:** Render city lights on the dark side of planet surfaces where available. May degrade performance.
- **Night light level:** Defines the brightness of night city lights. Valid range is 0 to 1. (ignored if planet night lights are disabled)

General effects

- **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.
- **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).
- **Reentry flames:** Render glowing plasma hull during reentry.
- **Particle streams:** Render ionised exhaust gases and vapour trails with particle effects. May degrade performance.

NEW!

3.4 Modules tab

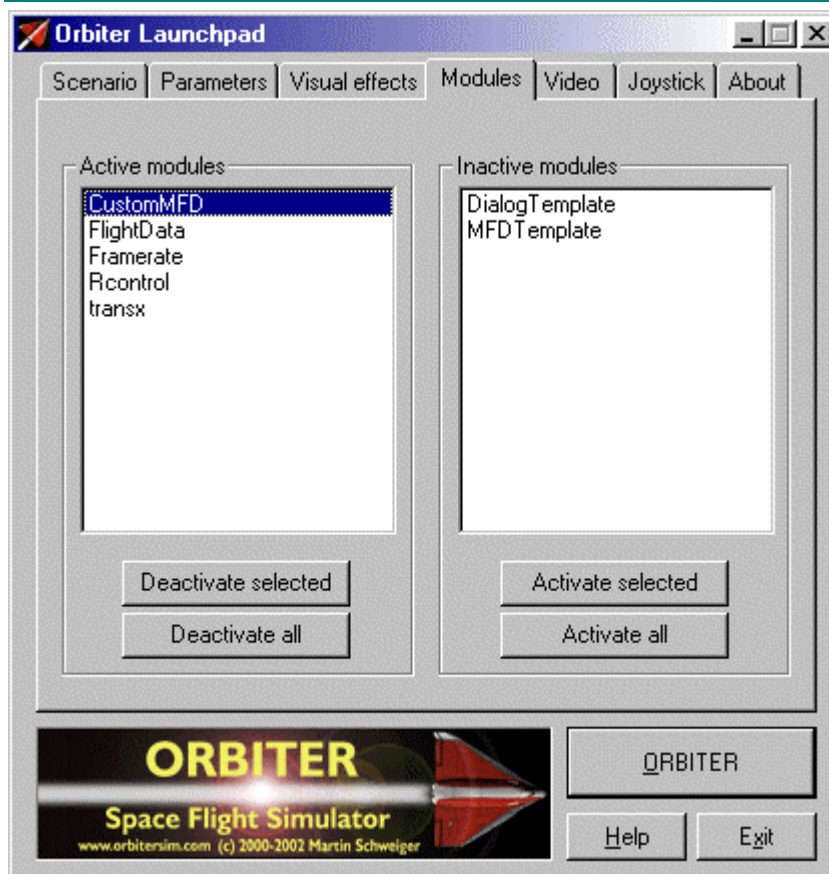


Figure 4: Launchpad dialog, Modules tab

This tab allows to activate and deactivate plugin modules for ORBITER which can extend the functionality of the core simulator. The modules provided with the standard ORBITER distribution are demos from the SDK package, and are available in full source code. Additional modules may be produced by add-on developers.

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

3.5 Video tab

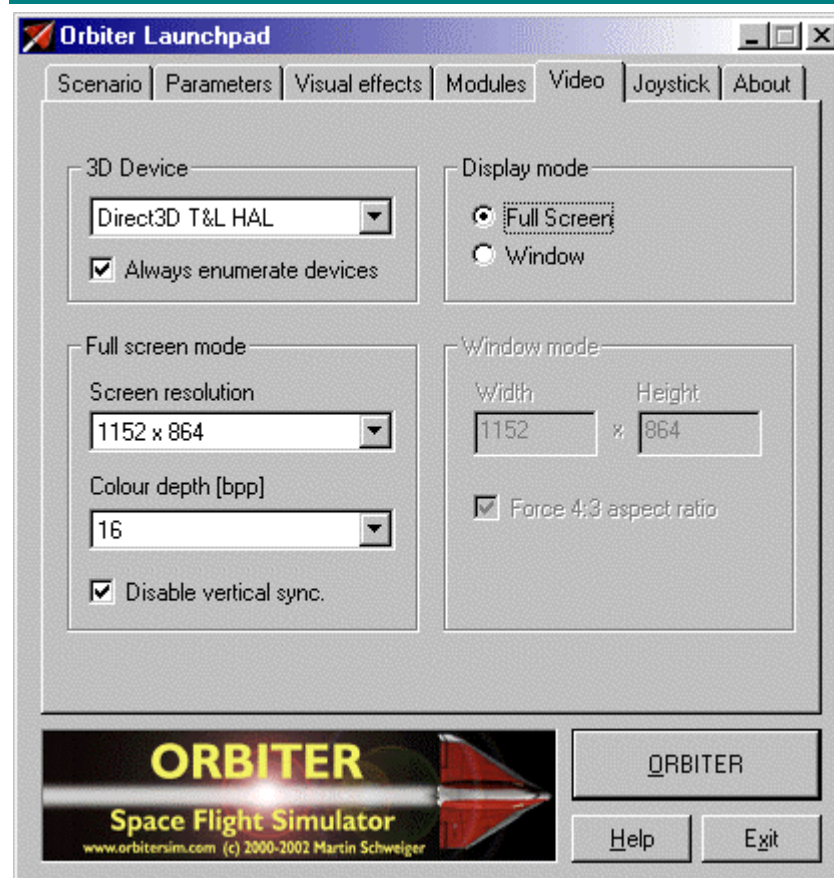


Figure 5: Launchpad dialog, Video tab

3D Device: Lists the available hardware and software devices for 3D rendering. Select a hardware device when possible, such as Direct3D HAL or Direct3D T&L HAL. 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.**

Display mode: Select fullscreen or windowed mode. Not all devices support windowed mode. Running in fullscreen mode may improve performance.

Fullscreen mode: Select screen resolution and colour depth for fullscreen mode. 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.

NEW!

Disable vertical sync: Ticking this box allows Orbiter to update a frame without waiting for a sync signal from the monitor. This can improve frame rates in fullscreen mode, but may lead to visual artefacts (tearing).

Window mode: Select the size of the render window for windowed modes. Select a width:height ratio close to 4:3 for best results. Large window sizes will reduce performance.

3.6 Joystick tab

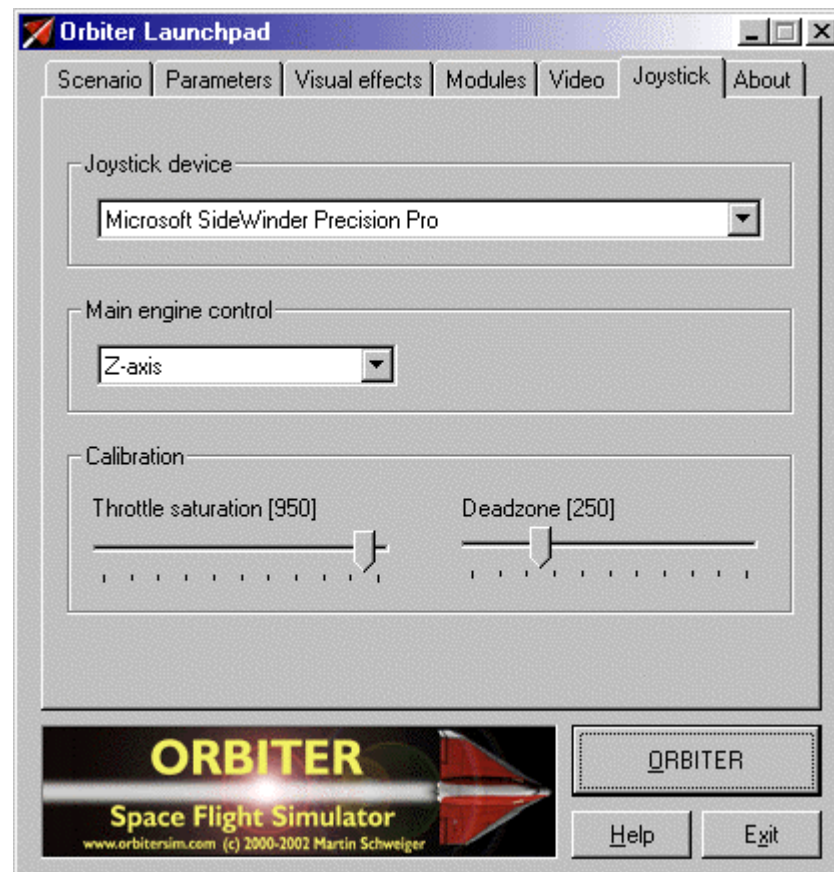


Figure 6: Launchpad dialog, Joystick tab

Joystick device: Lists all attached joysticks.

NEW!

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

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 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 5 and 6 on keyboard and joystick interface, section 11 on instrumentation, section 12 on spacecraft controls, and section 14 on basic flight manoeuvres.

Make sure you have configured Orbiter before launching your first simulation, in particular the video and joystick parameters (see section 3).

Starting:

- Select the *New Features/Atmospheric flight/Deltaglider runway takeoff* scenario (see Section 3.1 on scenario selection), and press the "ORBITER" button to launch the scenario. You are in control of a powerful spacecraft ready for takeoff at runway 33 of the SLF (Shuttle Landing Facility of the Kennedy Space Center).
- You can always exit the simulation by pressing **Ctrl** **Q** or **Alt** **F4**, or by clicking "Exit" on the main menu (**F4**). Orbiter always 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, if you have a joystick with a direction controller ("coolie hat"), you can use that to rotate the camera.
- To jump into the cockpit of your glider, press **F1**. (**F1** always toggles between cockpit and external view of the spacecraft you are controlling).

Instrument panels:

You should now see the instrument panel of your glider in the lower half of the simulation window.

- The panel can be scrolled with the cursor keys (**↓** **↑** **→** **←**) 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.
- You can remove the panel altogether and instead bring up a "generic" instrument view by pressing **F8**. Some spacecraft types don't come with custom panels – they only offer the generic view.
- Some spacecraft have more than a single panel which can be accessed by pressing **Ctrl** in combination with a cursor key. Bring back the panel view with **F8** (if you haven't already done so). Now press **Ctrl** **↑**. This will bring up 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** **↑**.

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 CRT 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 are always **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.

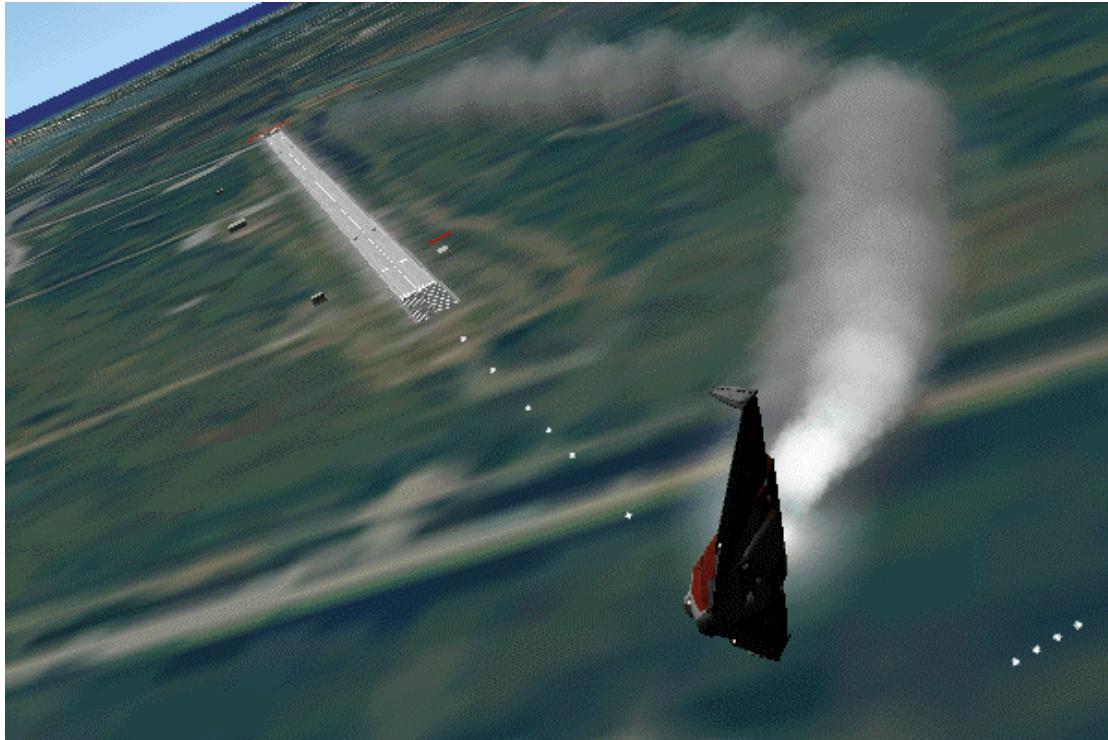
- Click on one of the buttons along the left or right edge to select the corresponding mode. If you click the top-left button, the MFD switches to *Orbit* mode.
- Modes can also be selected directly with a **[Shift]** key combination. The key combinations used for selecting a mode are listed in grey below the modes. (For example, **[Shift]** **[O]** will select the Orbit mode.)
- Most modes 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 labelled “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]** **[F]**).
- For now, switch the left MFD into *Surface* mode, and the right MFD into *HSI* mode.



Takeoff:

Your glider is capable of runway takeoffs and landings (provided the atmospheric density is sufficient to provide aerodynamic wing forces)

- For takeoff, engage main engines at full thrust. You can do this by pushing the *Main* engine sliders at the right of the panel to the top (make sure you push both sliders simultaneously!), or by pressing **[Ctrl]** **[+]** *Numpad*. 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, pull back on the joystick to rotate, or press and hold **[2]** *Numpad*.
- 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]** *Numpad* key until hover engines are fully engaged.
- Your glider should now lift off vertically. Once clear of the ground, engage main engines.
- 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 feel for handling at different altitudes. Without a joystick, you can use the numerical keypad (**[2]**/**[8]** *Numpad* for pitch, **[4]**/**[6]** *Numpad* for roll, and **[1]**/**[3]** *Numpad* 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 you 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 11.4 for details.
- As you approach the runway, you will see PAPI and VASI landing aids in front of and beside the runway (see section 14.6). 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 (**[.]** and **[.]**) 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 on 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 ⊕ 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 left side of the instrument panel) or by pressing  . 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 grey circle), together with a list of orbital parameters along the left side of the display. You should switch the display into “current orbital plane” projection mode, by clicking on the “PRJ” button until “Proj: Ship” is displayed in the top right corner of the display.
- 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 distance (“ApD”). 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 distance (the highest point of the orbit) will start to grow. Keep firing your engines until ApD reaches about 6.670M. This corresponds to an altitude of 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  twice. In this mode, the HUD flight path ladder is aligned with the orbital plane instead of the horizon plane. It also shows indicators for prograde (the direction of your orbital velocity vector) and retrograde (the opposite direction).
- When you approach apoapsis (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  to engage time acceleration, and  to switch back), turn your craft prograde. You can do this manually, but it is easier to leave it to the automatic attitude control, by simply pressing the “Prograde” button on the left of the instrument panel.
- Now fire your main engines for final orbit insertion. The two parameters to watch are the orbit eccentricity (“Ecc”) and periapsis distance (“PeD”). The eccentricity value should get smaller, indicating that the orbit becomes more circular, while the periapsis distance approaches the apoapsis distance (ApD). 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, 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. Reentries are usually performed with a high angle of attack (AOA) – about 40° 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 16. First you might want to learn a bit more about orbital manoeuvres and docking procedures in section 14.

5 Keyboard interface

The key assignment reference in this section and the rest of the manual refers to the keyboard layout shown in Figure 7. 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.

 Numpad Or  Cursorpad.

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

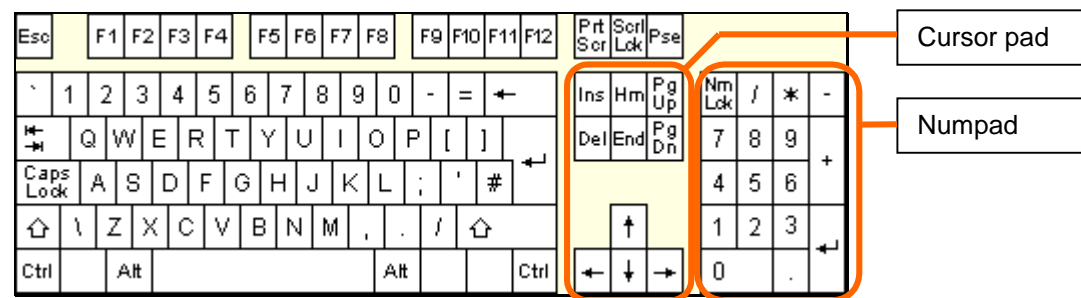





































Figure 7: Keyboard layout reference

5.1 General





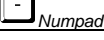
	Toggle between frame rate and field-of-view display.
	Time warp shortcut: Slow down simulation by factor 10 (down to real-time). See also Time acceleration dialog ( )
	Time warp shortcut: Speed up simulation by factor 10 (up to a maximum warp factor of 10000). See also Time acceleration dialog ( )
	Zoom out (increase field of view). See also Camera dialog ( )
	Zoom in (decrease field of view). See also Camera dialog ( )
 	Undock from a vessel. (In conjunction with an old-style orbital station: Request docking or undocking clearance if within radio range of the selected station)
 	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.
	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.
	Switch control to a different spacecraft.
	Main menu.

	Open the <i>Custom functions</i> dialog. Contains a list of functions defined in plugin modules, if available.
	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.
	In cockpit view, toggle between onscreen instruments and custom panels (if supported by the spacecraft).
	“Planetarium mode”: Toggle display of constellations.



5.2 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.



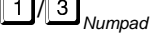

Main/retro thruster controls:


	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):


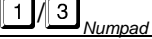
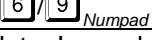
	Increase hover thruster setting.
	Decrease hover thruster setting.


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)
	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):

	Engage attitude thrusters for up/down translation.
	Engage attitude thrusters for left/right translation.
	Engage attitude thrusters for forward/back translation









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












Other controls:

	Toggle reaction control thruster mode between rotational (engage opposite thruster pairs) and linear (engage parallel thruster pairs)
---	---

  <i>Numpad</i>	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.
  <i>Numpad</i>	Enable/disable manual user control (via keyboard or joystick) of aerodynamic control surfaces (elevator, rudder, ailerons) if available.
	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.
	Toggle “H-level” navcomp mode. This mode keeps the spacecraft level with the horizon by engaging appropriate attitude thrusters.
	Toggle “Turn prograde” navcomp mode. This mode turns the spacecraft into its orbital velocity vector.
	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 antinormal to its orbital plane (in the direction of $-\vec{R} \times \vec{V}$)
  <i>Cursorpad</i>	Trim control (only vessels with aerodynamic surfaces)
	Apply left wheel brake (where applicable)
	Apply right wheel brake (where applicable)

5.3 External views













	Toggle display of information about current object.
	Move camera away from target object.
	Move camera towards target object.
    	Rotate camera around object.

In ground-based camera views,      will move the observer position,  and  will change the observer altitude, and     will rotate the observer direction (unless locked to the target).

5.4 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.

	Map between onscreen MFD mode and instrument panel mode (if available for the current spacecraft)
   	Scroll instrument panel.
    	Switch to neighbour panel, if available.
 	Toggle HUD display on/off.

	Switch HUD mode.
	HUD reference selection. Orbit HUD: opens reference selection input box. Docking HUD: steps through available NAV receivers.
	Docking HUD: Reference selection. Can also be used for legacy orbital stations which do not define XPDR and IDS transmitters.
	Open a menu for left/right MFD mode selection.
	Open/page/close the MFD-specific parameter selection menu.
	Display <i>Align orbital plane</i> mode MFD.
	Display <i>Docking</i> mode MFD.
	Display <i>Launch/landing</i> mode MFD.
	Display <i>Map</i> mode MFD.
	Display <i>Orbit</i> mode MFD.
	Display <i>Surface</i> mode MFD.
	Display <i>Transfer</i> mode MFD.
	Display <i>Synchronise orbit</i> mode MFD.
	Turn off MFD.

Docking Mode MFD

	Input new docking target.
--	---------------------------

Align orbital plane Mode MFD

	Input new target object.
--	--------------------------

Orbit Mode MFD

	Auto-select reference object.
	Toggle display mode (list only, graphics only and both)
	No target orbit.
	Toggle orbit projection mode (ecliptic, ship's and target's orbital plane)
	Select new reference object (planet or moon) for orbit calculation.
	Open menu for target selection.

Map Mode MFD

	Open input box for reference planet/moon selection.
	Open a menu for target selection.

Transfer Mode MFD

	Open input box for reference planet/moon selection.
	Open a menu for source orbit object selection.
	Open a menu for target selection.
	Unselect target.
	Toggle (hypothetical) transfer orbit display on/off.

	Toggle numerical multibody trajectory calculation.
	Refresh numerical trajectory, if displayed.
	Open input box for time step definition.
	Rotate transfer orbit ejection longitude.
	Decrease/increase transfer orbit major axis.

Synchronise orbit Mode MFD





	Input new target object (orbital station or moon)
	Change reference axis.
	Select number of entries in the reference transit time list.
	Rotate reference axis (in manual mode only).

5.5 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.
	Select current item and close list.
	Cancel list.

6 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 or 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   <i>Numpad</i> and   <i>Numpad</i> keyboard controls, but it affects only the main thrusters, not the retro thrusters.
Direction controller ("coolie hat")	External view: Rotate camera around the observed object Internal view: Scroll instrument panel (if applicable)

7 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.

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

8 Spacecraft classes

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

8.1 Delta-glider

The Delta-glider is the ideal ship for the novice pilot to get spaceborne. Its futuristic design concept 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 glider comes with operating landing gear, nose cone docking port, airlock door, deployable radiator and animated aerodynamic control surfaces. It now supports particle exhaust effects.

NEW!

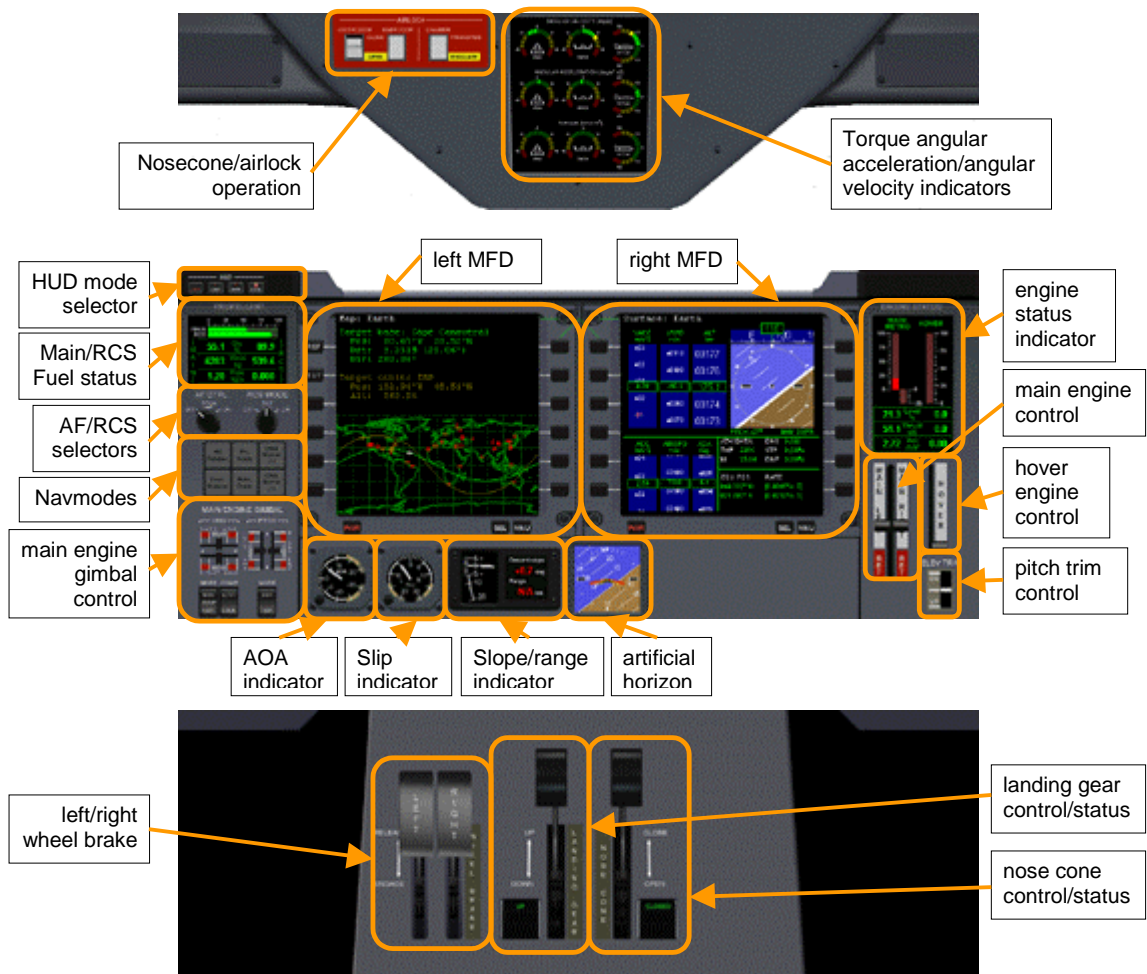
The new version makes use of the new atmospheric flight model for significantly more realistic aerodynamic performance.

Technical specifications:

Mass	12.0·10 ³ kg	(empty orbiter)
	11.4·10 ³ kg	(main fuel load)
	0.6·10 ³ kg	(RCS fuel load)
	24.0·10 ³ kg	(total)
Length	17.76 m	
Wingspan	17.86 m	
Thrust	2 x 1.2·10 ⁵ N	(main engines)
	2 x 2.7·10 ⁴ N	(wing-mounted retro thrusters)
	3 x 9.0·10 ⁴ N	(hover thrusters)
Isp	4·10 ⁴ m/s	(fuel-specific impulse in vacuum)
Inertia (PMI)	15.5 / 22.1 / 7.7 m ²	
Stall C _L	1.0	
Stall AOA	20°	

Main and overhead instrument panels:

Turn the panels on and off with **F8**. The glider supports three panels which can be selected with **Ctrl** **↓** and **Ctrl** **↑**.



RCS and aerodynamic control selection

NEW!

The AF CTRL selector is used to activate control of aerodynamic surfaces via manual user input. Manipulating the control surfaces is only effective within an atmosphere at sufficiently high dynamic pressures. The settings are: OFF (control surfaces offline), ON (control surfaces enabled), and PITCH (only pitch control enabled).



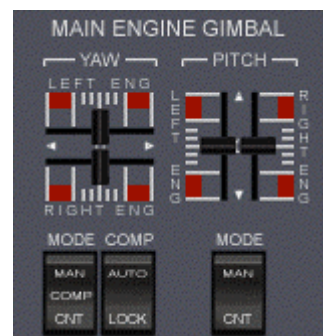
The RCS MODE selector sets the Reaction Control System mode which is used to control attitude in free space. During atmospheric flight, when aerodynamic control surfaces are active, the RCS is usually disabled. The selector can be set to OFF (RCS disabled), ROT (RCS in rotational mode), and LIN (RCS in linear mode).

Both selectors can be set with the left and right mouse buttons. Shortcuts for RCS are **[Numpad]** (ROT/LIN) and **[Ctrl][Numpad]** (ON/OFF). Shortcut for AF control is **[Alt][Numpad]** (ON/OFF).

Main engine gimbal control






Both main engines can be gimballed independently in pitch and yaw. Gimbal range is $\pm 1.0^\circ$ in pitch, and $\pm 7.7^\circ$ in yaw. The yaw range allows to compensate for torque generated by a single engine at main thrust.

To use pitch gimbaling, the mode selector must be set to "MAN". The gimbal sliders can then be dragged with the mouse. "CNT" re-centers the pitch gimbals.



The yaw gimbal selector has three modes: “MAN” for manual slider control, “CNT” for re-centering gimbals, and “COMP” for *compensation* mode. There are two compensation modes available: With “LOCK”, both engines are set to divergent thrust at their extreme limit, so that both thrust vectors are aligned with the glider’s centre of gravity; with “AUTO”, both engines are set to parallel thrust, and the gimbal angle is adjusted so that the total thrust vector is aligned with the centre of gravity.

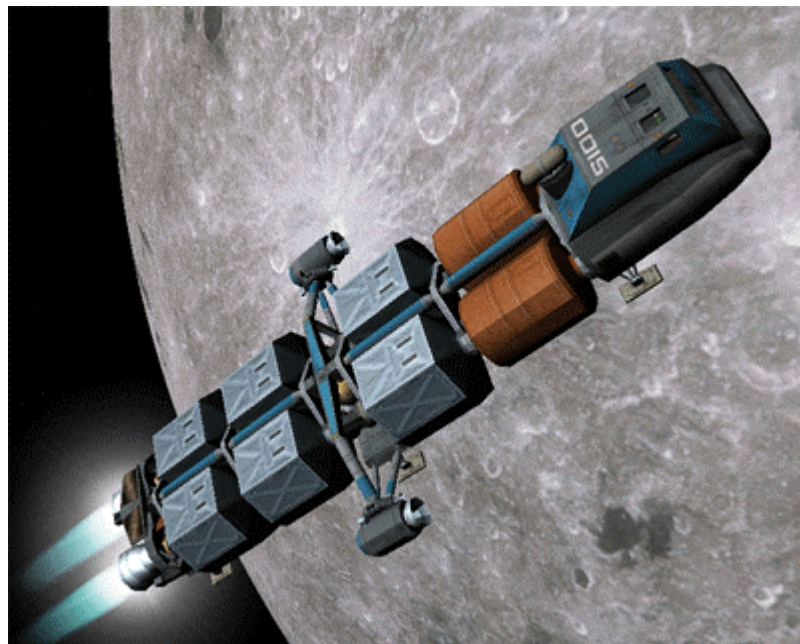
Vessel-specific key controls:

	Operate landing gear
	Operate nose cone docking mechanism
	Open/close outer airlock door
	Deploy/retract airbrakes
	Deploy/retract radiators

8.2 Shuttle-A

The all-new Shuttle-A, designed by Roger “Frying Tiger” Long. A 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.

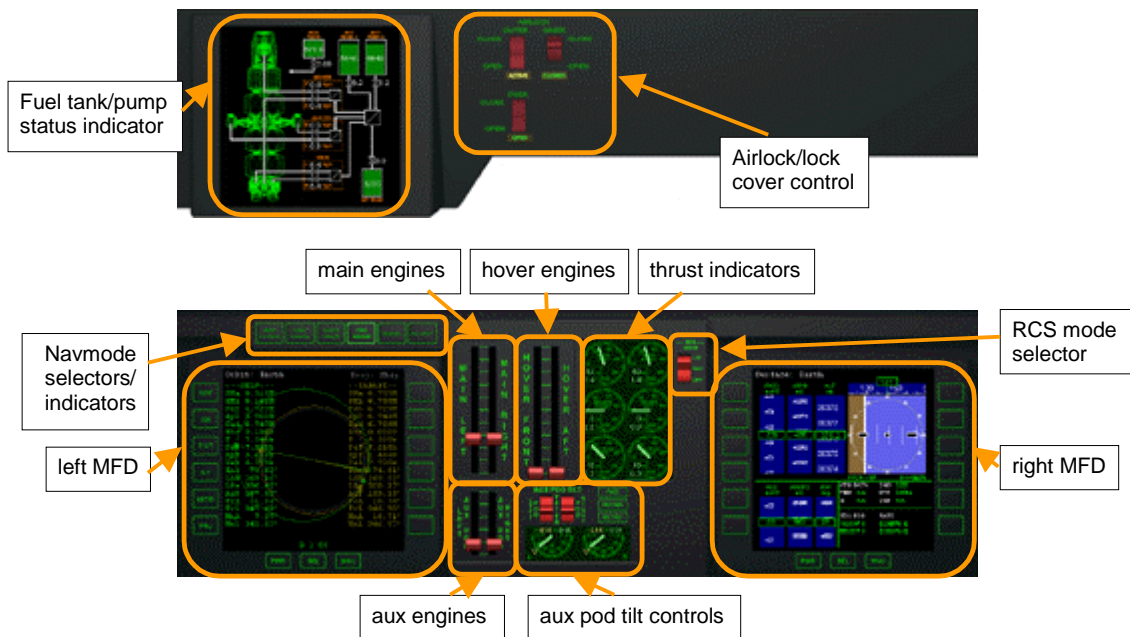


Model design: Roger Long. Instrument panels and module code: Martin Schweiger.

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

Main and overhead instrument panels:

Turn the panels on and off with . The Shuttle-A supports two panels which can be selected with  and .



Vessel-specific key controls:

	Operate docking hatch mechanism
	Open/close outer airlock door

8.3 Shuttle PB (PTV)



Mass: $2 \cdot 10^3$ kg (empty), $5 \cdot 10^3$ kg (full)
 Length: 7 m
 Thrusters: $1.2 \cdot 10^5$ N (main)
 $4.0 \cdot 10^4$ N (retro)
 $8.0 \cdot 10^4$ N (hover)

Very agile single-seater. Produces little lift in atmospheric flight.

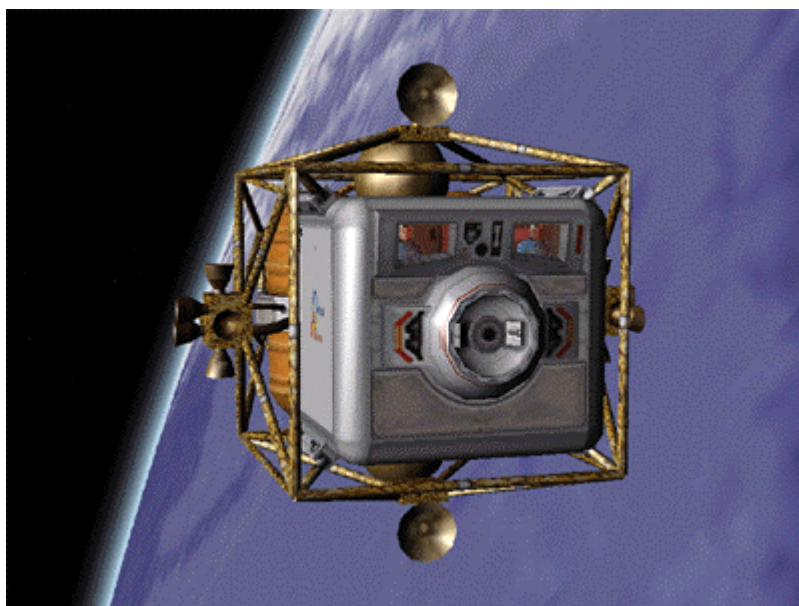
Overall design and textures: Balázs Patyi. Model improvements: Martin Schweiger

8.4 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 modelled with detailed electrical and environmental systems simulation, contributed by Radu Poenaru. For detailed information see the Dragonfly Operations Handbook.

Technical specifications:

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

8.5 Space Shuttle Atlantis

The first “real” spacecraft to appear in the Orbiter standard distribution. Launch configuration consists of Orbiter, main tank and 2 solid rocket boosters (SRB). The latest Orbiter distribution contains a new Atlantis model by Don Gallagher, and external tank and SRB models by Damir Gulesich.

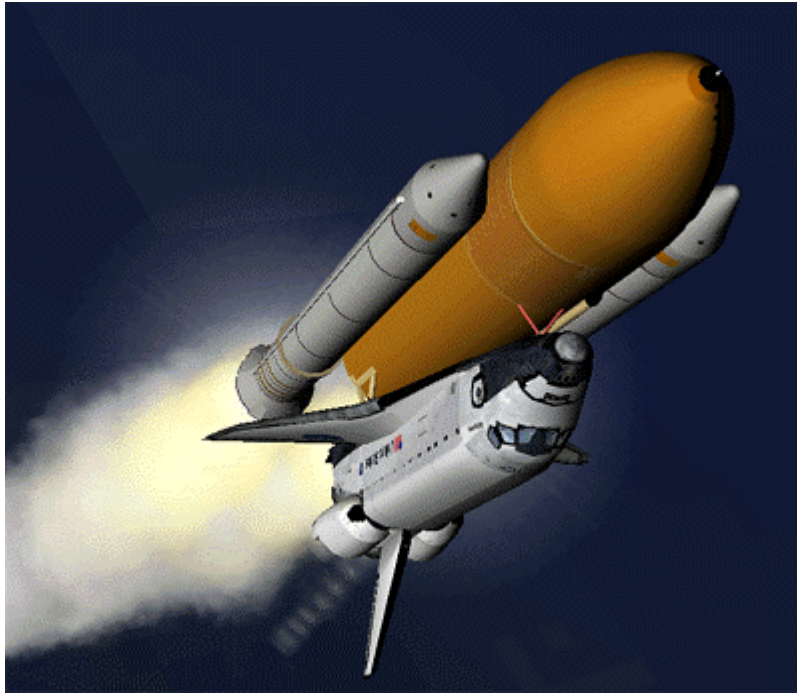
NEW!

Atlantis features a functional arm with grappling capabilities and MMU support, contributed by Robert Conley. See *Atlantis_MMU_Sat* manual for details.

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.
- During launch, attitude is controlled via SRB thrust vectoring. Roll shuttle for required heading, and decrease pitch during ascent for required orbit insertion.
- 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.



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

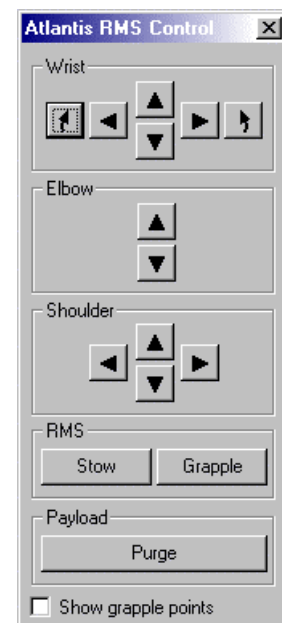
Docking:

- 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.

NEW!

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.

Technical specifications:

Orbiter	
Length	39.16 m
Wingspan	24.54 m
Height	14.29 m
Mass	2041166 kg (liftoff)
	104326 kg (end of mission)
Max. cargo	28803 kg
SME thrust	$6.26 \cdot 10^6$ N (vac.)
	$5.01 \cdot 10^6$ N (liftoff)
OMS thrust	$5.34 \cdot 10^4$ N (vac.)

Tank^b	
Length	47.83 m
Diameter	9.68 m
Mass	35425 kg (empty)
	719115 kg (propellant)
	756445 kg (total)

SRB	
Length	45.7 m
Diameter	3.8 m (tube)
	5.9 m (max)
Mass	87543 kg (empty)
	502126 kg (propellant)
	589670 kg (total)
Thrust	11791820 N (liftoff)

Orbiter+Tank assembly	
Length	57.55 m
Height	24.44 m

Orbiter+Tank+SRBs (launch assembly)	
Length	57.91 m
Height	24.44 m

^a: principal moments of inertia tensor, mass-normalised, assuming homogeneous density distribution

^b: including Orbiter mount brackets

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.



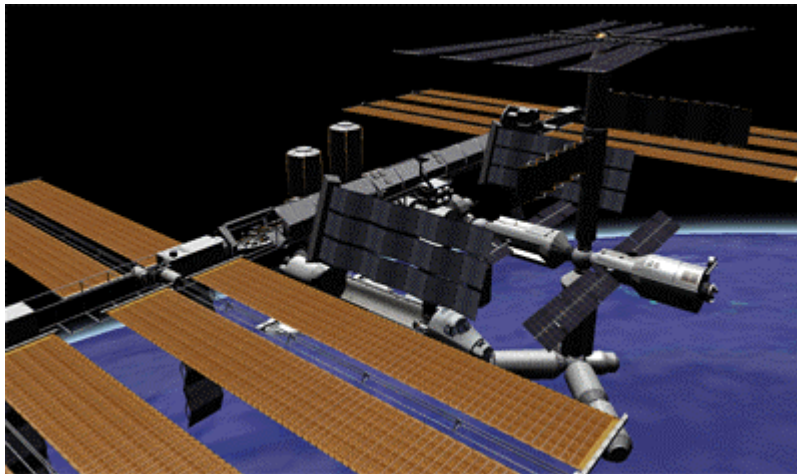
Unlike the futuristic spacecraft designs, Atlantis provides only a small margin of error for achieving orbit. Try some of the other ships before attempting to launch the Shuttle. Limited fuel *must* be selected, otherwise Atlantis is too heavy to reach orbit!

8.6 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.

NEW!

Orbiter now includes Andrew Farnaby's *Project Alpha* ISS model.



3D model and textures
by Andrew Farnaby

In Orbiter, the ISS can be tracked with its transponder (XPDR) signal, which by default is set to frequency 131.30.

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 14.7.

8.7 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 manoeuvre.

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

Port 1	135.00
Port 2	135.10
Port 3	135.20

8.8 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 m/s^2 , or about 0.8g, to mimic Earth's surface gravitational force.

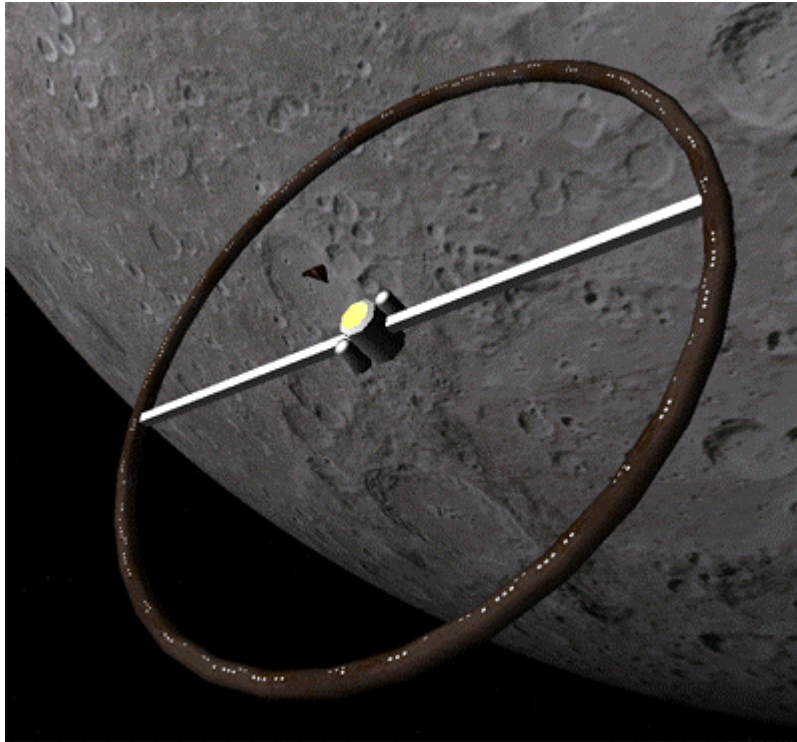
The main problem the station poses to the spacecraft pilot is in performing a docking manoeuvre. Docking to a rotating object is only possible along the rotation axis. The wheel has two docking ports in the central hub. The docking approach is performed along the axis of rotation. Before docking, the approaching vessel must synchronise its own longitudinal rotation with that of the station. For docking procedures, see Section 14.7.



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







8.9 Hubble Space Telescope

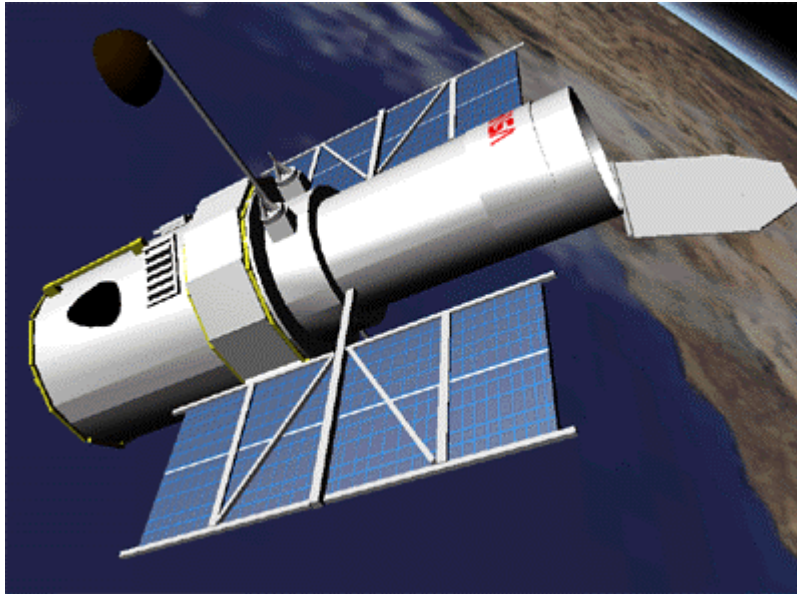
The Hubble Space Telescope is the visible/ultraviolet/near-infrared element of the Great Observatories astronomical program. The spacecraft provides an order of magnitude better resolution than is capable from ground-based telescopes. The objectives of the HST are to: (1) investigate the composition, 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.

The Hubble Space Telescope (HST) is a space-borne deep-space observation platform. Originally deployed with a flawed primary mirror, it has undergone several repair and refitting operations to its optical system and instrumentation, and is now one of the most successful telescopes.

Orbiter provides several Space Shuttle/HST missions for both deployment and recapture operations. For Shuttle payload manipulation, see Section 8.5 above.

HST-specific key controls:

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



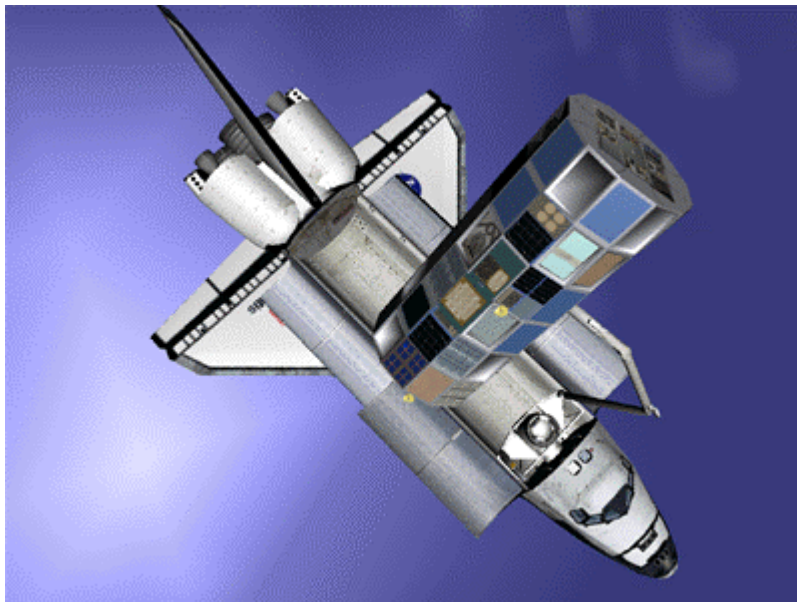
HST model and textures by David Sundstrom

8.10 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 January 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.

9 Camera modes

To open a camera configuration dialog, press **Ctrl F1**. This allows to

- Select the camera target in external views.
- Jump back to the current focus object in external or cockpit view. (Shortcut: **F1**)
- Select the external camera tracking or ground-based mode. (Shortcut: **F2**)
- Change the camera field of view (FOV). (Shortcut: **Z** and **X**)
- Store and recall camera modes via the preset list.

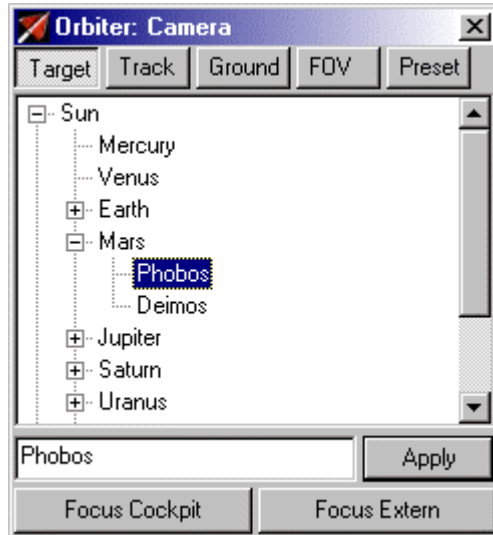


Figure 8: Camera dialog – target selection list.

9.1 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.

To switch between full instrument panel and MFD-only display modes press **F8**. Note that instrument panels may not be available for all spacecraft. Among the standard Orbiter vessel types, only the Deltaglider and Shuttle-A currently support panels.

Panels can be scrolled with **↓** **↑** **→** **←**. If the spacecraft supports multiple panels, they can be selected with **Ctrl** **↓** **↑** **→** **←**.

For details on HUD and MFD modes, see sections 10 and 11.

9.2 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. 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. 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. 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 () is deactivated, but radial camera movement with and is still available.

NEW!

Ground-based views place the camera at a fixed point on 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. Pick an observer location by selecting the planet, location, and spot from the lists, and click *Apply*. You can move the observer location by pressing , and the observer altitude by pressing and . 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 .

See also Section 17.3 on how to add new observer sites to a planet definition file.



Figure 9: Selecting a track camera mode (left) or a ground-based view (right).

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

9.3 Selecting the field of view

The camera aperture can be adjusted under the *FOV* tab in the Camera dialog. The supported range is between 10° and 90° (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° and 60° .

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.

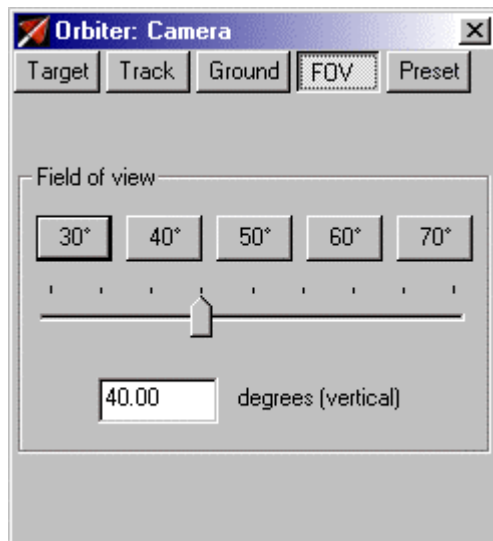


Figure 10: Camera field of view selection.

The keyboard shortcuts are **[Z]** to decrease the FOV, and **[X]** to increase the FOV. The current field of view is displayed in the status section in the top left corner of the simulation window.

9.4 Storing and recalling camera modes

NEW!

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.



Figure 11: The camera mode preset 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.

10 Head-up display

The Head-up display (HUD) provides information about the current status of your spacecraft when internal (cockpit) view mode is selected. 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 11).

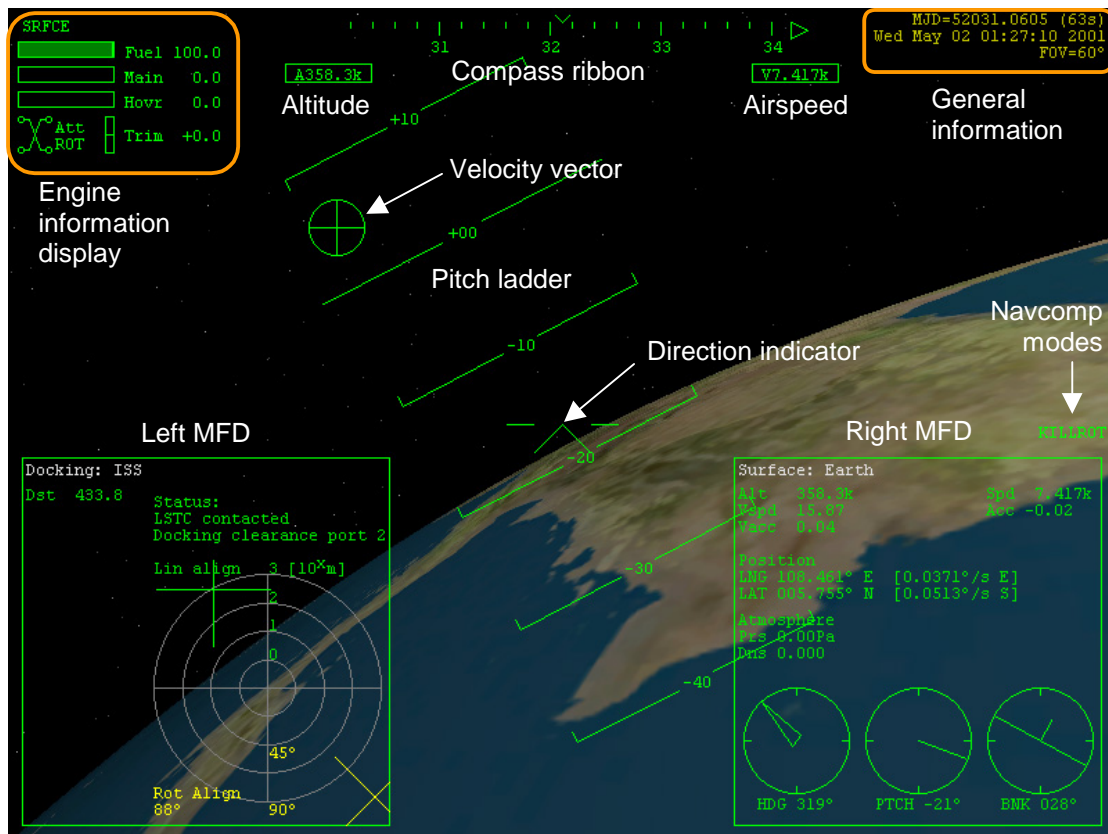


Figure 12: HUD in surface mode, including docking and surface MFDs.

10.1 General information display

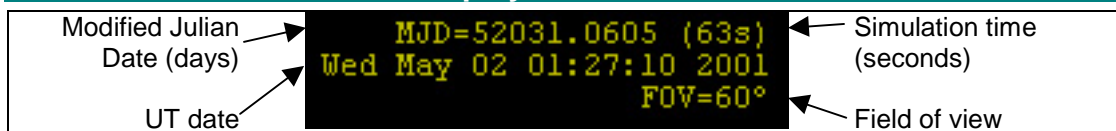


Figure 13: General simulation information

MJD: The *Julian Date* (JD) is the interval of time in mean solar days elapsed since 4713 BC January 1 at Greenwich mean noon. The *Modified Julian Date* (MJD) is the Julian Date minus 240 0000.5

UT: Universal time counted from 0^h at midnight. Unit is mean solar day.

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

The display can be toggled between FOV and viewport resolution/frame rate via the **[F]** key.

10.2 Engine information display

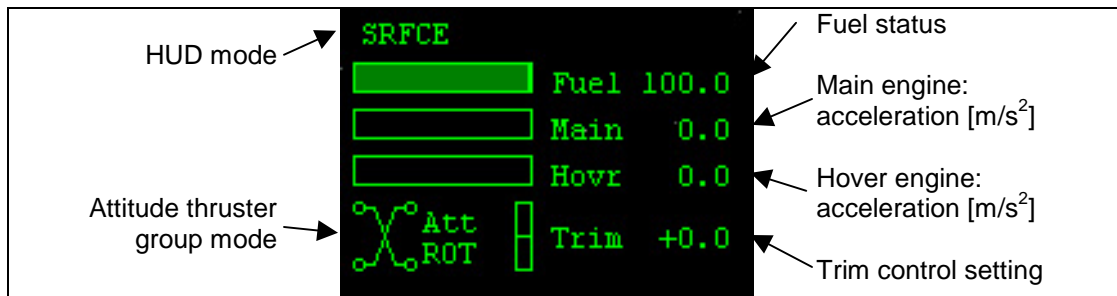


Figure 14: Fuel/engine displays

The engine information display is only shown in non-panel 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^2 (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.

Attitude thruster mode: Attitude thrusters are small engines used for rotation and fine translational adjustments. The display shows the current mode (rotational/translational).

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

For more information about engines and spacecraft control see Section 12.

10.3 Surface 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 "⊕".

10.4 Orbit 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 "0" 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 Ctrl R.

10.5 Docking 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 "⊕". 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 ⊕ nor + are visible, then the direction of the ⊕ 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  .

11 Multifunctional display modes

Two multifunctional displays (MFD) can be displayed simultaneously on the screen. MFDs provide essential information about all spacecraft flight parameters. A range of different MFD modes is available to assist in various navigational problems such as launch/landing, surface flight, orbit determination etc. MFDs are controlled by Shift key combinations, where the left Shift key controls the left MFD, the right Shift key controls the right MFD.

MFD modes can be selected either directly (see section 5.4 for keyboard commands) or by pressing **[Shift][F1]** to bring up a menu.

Each MFD mode may support specific parameters which can be selected with the appropriate Shift key combination. You don't need to remember the key codes, since MFDs which allow parameter selection provide a menu mode which can be selected with **[Shift][I]**. Pressing **[Shift][I]** again will switch to the next menu page or return to the MFD display. To select a menu item, press **[Shift]** with the highlighted character key.

11.1 COM/NAV receiver setup

The *COM/NAV setup* MFD mode provides an interface to the ship's navaid signal receivers which feed data to the navigation instruments. The mode is activated by **[Shift][C]** or via the *Comms* entry from the MFD mode menu (**[Shift][F1]**).

The MFD lists frequency and signal source information for all NAV radios (NAV1 to NAV n).

The number of receivers (n) depends on the vessel class. A NAV receiver can be selected by **[Shift][,]** and **[Shift][.]**. The selected receiver is highlighted in yellow (see below).

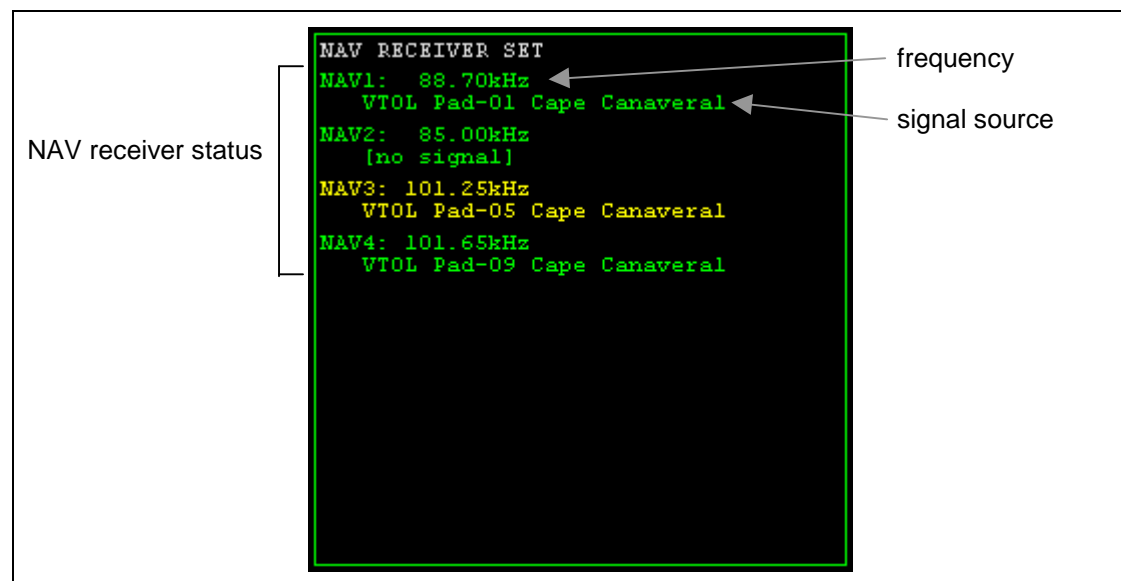


Figure 15: NAV setup MFD

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

Key options:

[Shift][,]	Select previous NAV receiver
[Shift][.]	Select next NAV receiver
[Shift][=]	Step down frequency 1 kHz.
[Shift][I]	Step up frequency 1 kHz
[Shift][I]	Step down frequency 0.05 kHz

Shift I	Step up frequency 0.05 kHz
-----------------------	----------------------------

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**) and *Navaid Info* (**Ctrl** **N**) are a useful tools to obtain frequencies for navaid transmitters such as VOR and ILS beacons.

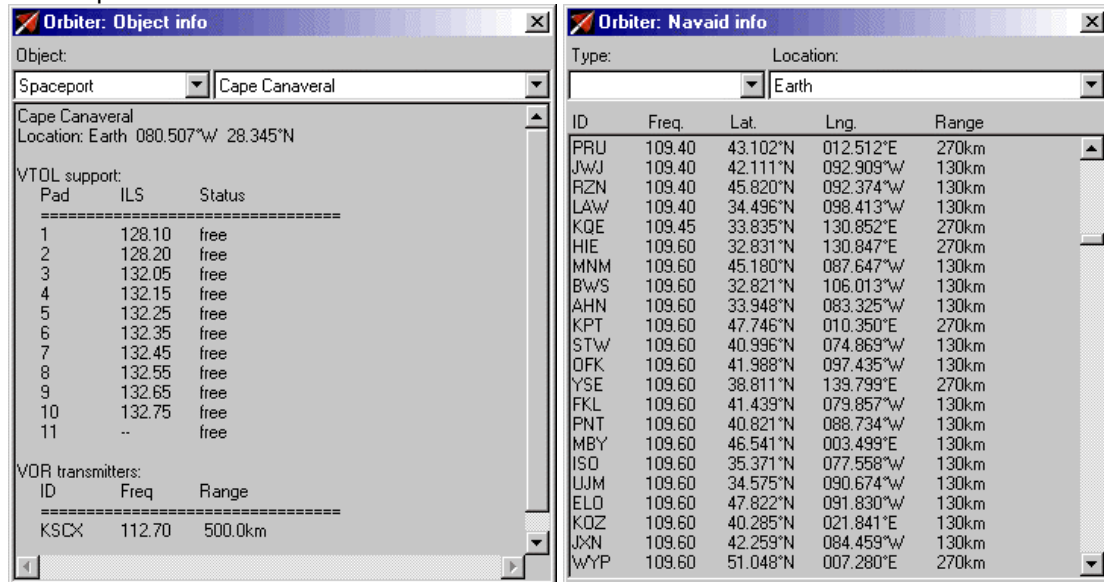


Figure 16: Info and Navaid dialogs with VOR and ILS frequencies.

11.2 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 parameters will then be displayed as well.

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**.

The plane into which the orbits are projected can be selected via **Shift** **P**. The current projection plane is indicated in the top right corner of the instrument. *Ecliptic* projects into the plane of the ecliptic (Earth's orbital plane). *Ship* projects into the vessel's current orbital plane, and *Target* projects into the target's current orbital plane, if a target is specified.

Key options:

Shift A	Auto-select reference object.
Shift M	Toggle display mode (list only, graphics only and both)
Shift N	No target orbit.
Shift P	Toggle orbit projection mode (ecliptic, ship's and target's orbital plane)
Shift R	Select new reference object (planet or moon) for orbit calculation.
Shift T	Open menu for target selection.

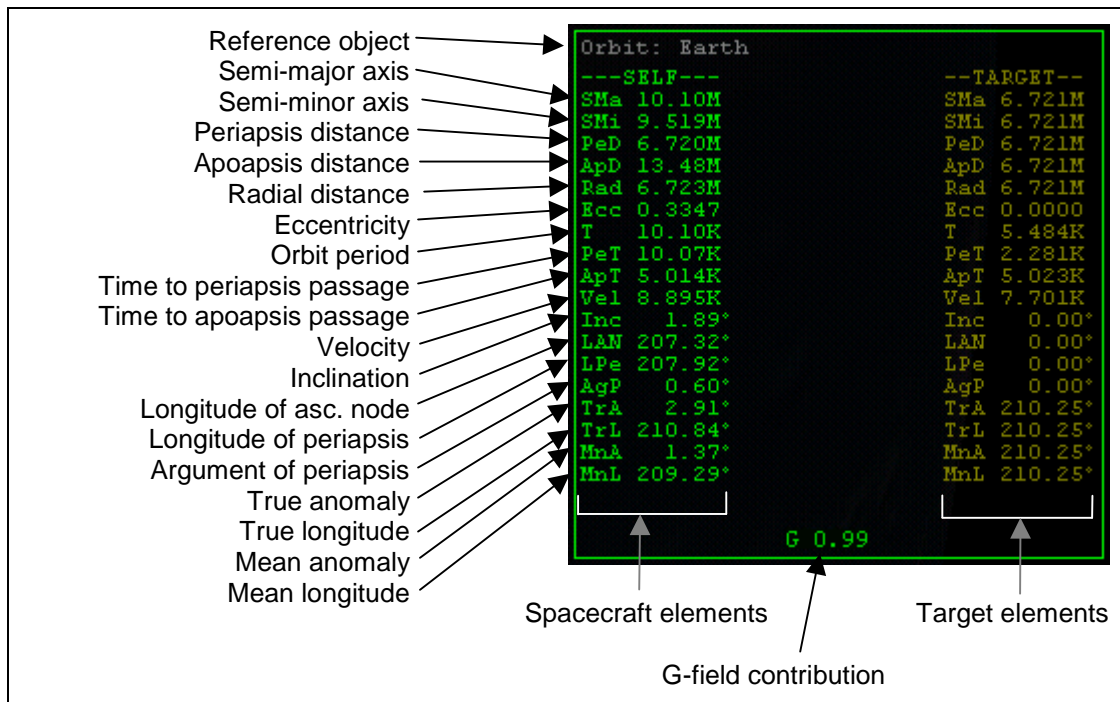


Figure 17: Orbit MFD in parameter list mode.

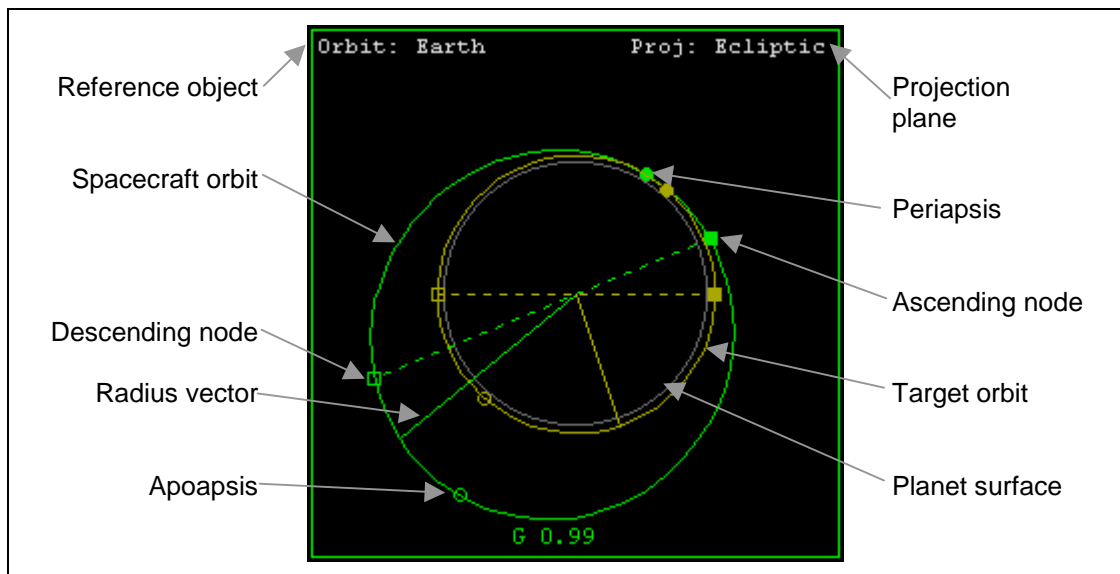


Figure 18: Orbit MFD in graphic mode.

Notation:

- Periapsis: The point of the orbital ellipse closest to the central body.
- Apoapsis: The point furthest away from the central body.
- Ascending node: The point at which the orbit passes through the ecliptic (or other reference plane) from below.
- Descending node: The point at which the orbit passes through the ecliptic from above.
- Radius vector: The vector from the central body to the current position of the orbiting body.

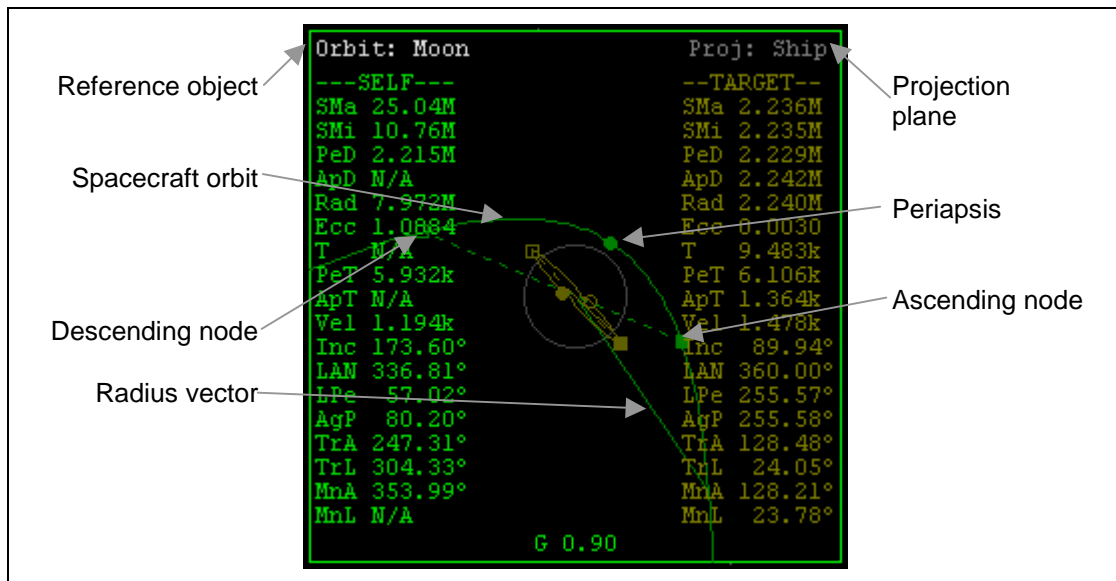


Figure 19: Hyperbolic orbit around the Moon

For hyperbolic orbits, the following parameters are interpreted specially:

SMa: real semi-axis a : distance from coordinate origin (defined by intersection of hyperbola asymptotes) to periapsis

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

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 $e \sinh H - H$, with H 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.

For further explanation of orbital elements see Appendix B.

11.3 VOR/VTOL

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

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. 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 NAV MFD mode (see section 11.1).

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.

- **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 ± 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.

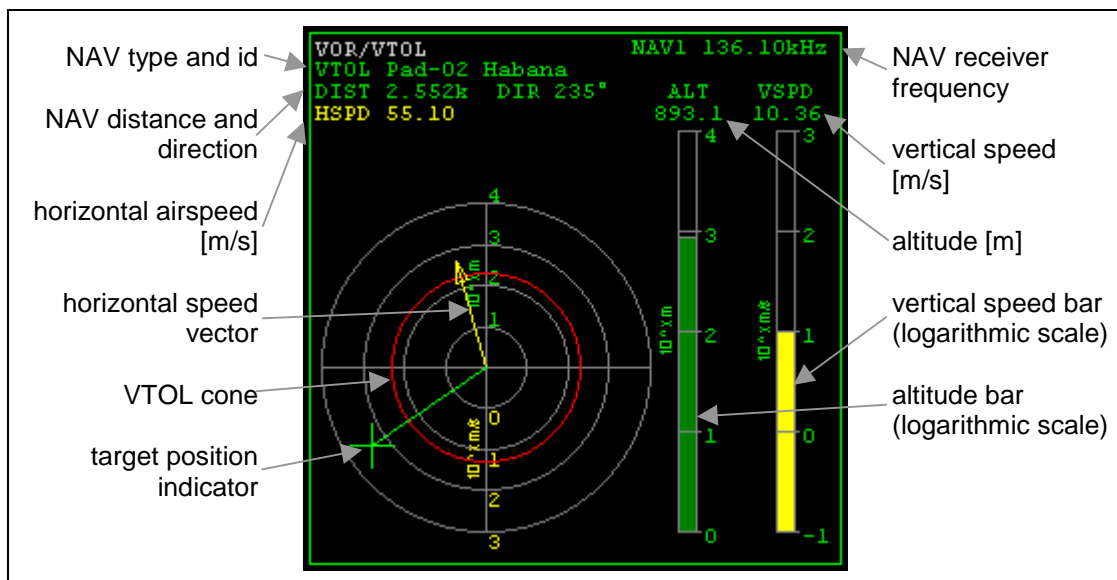


Figure 20: VOR/VTOL MFD.

Key options:

[Shift] [N]	Select NAV receiver
--------------------	---------------------

11.4 Horizontal Situation Indicator

NEW!

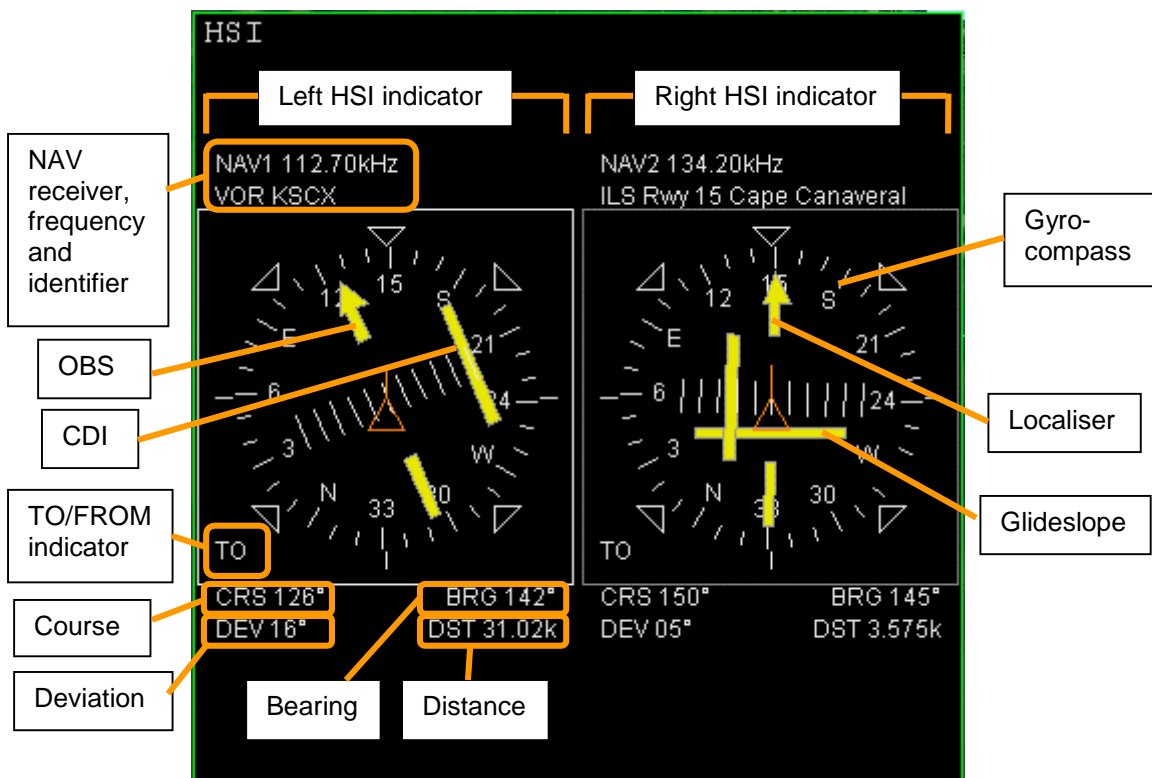
The Horizontal Situation Indicator (HSI) consists of two independent displays. Each display can be slaved to a NAV receiver and show directional and relative bearing information. The instruments accept data from surface-based transmitters such as VOR and ILS. The function is similar to instrument navigation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





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, **Shift** **C**)
- Slave one of the HSI displays to that receiver with **Shift** **N**.
- To fly directly towards the station, turn the OBS setting 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,  ), 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.





Key options:

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

11.5 Docking

The *Docking* MFD assists during final approach to dock with another vessel or orbital station. The layout is similar to the Landing MFD.

Docking approach data can be acquired in two 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  . 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  .

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

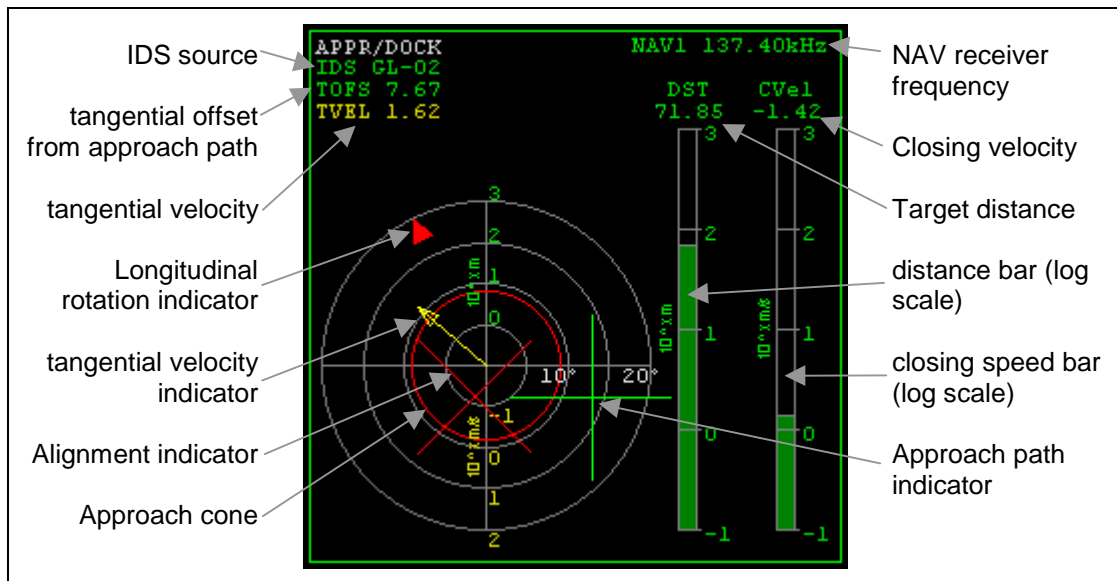








Figure 21: Docking MFD.

Key options:

 	Select NAV receiver for IDS information input.
 	Switch to visual data acquisition mode.
 	Input a new orbital station as docking target (obsolete option)

- **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 $0.1-10^3$ m.
- **CVEL:** Closing speed [m/s]. The bar shows the closing speed on a logarithmic scale in the range $0.1-10^3$ 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 $0.1-10^3$ m. Tangential alignment should be performed with attitude thrusters in *linear* mode (see Section 12.2).
- **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 $0.01-10^2$ 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 $0-20^\circ$. Rotational alignment should be performed with attitude thrusters in *rotational* mode (see Section 12.2).
- **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 12.2). 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° .

- **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:

- Engine controls are disabled while docked to an orbital station.
- 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.

11.6 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 ribbon**
- **Altitude indicator** with markers for perihel and aphel altitude
- **Vertical speed**
- **Vertical acceleration**
- **Airspeed**
- **Acceleration**
- **Angle of attack**
- **Atmospheric data**
- **Equatorial position** (longitude and latitude, and rate of change)

NEW!

The following atmospheric data are displayed (if applicable):

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

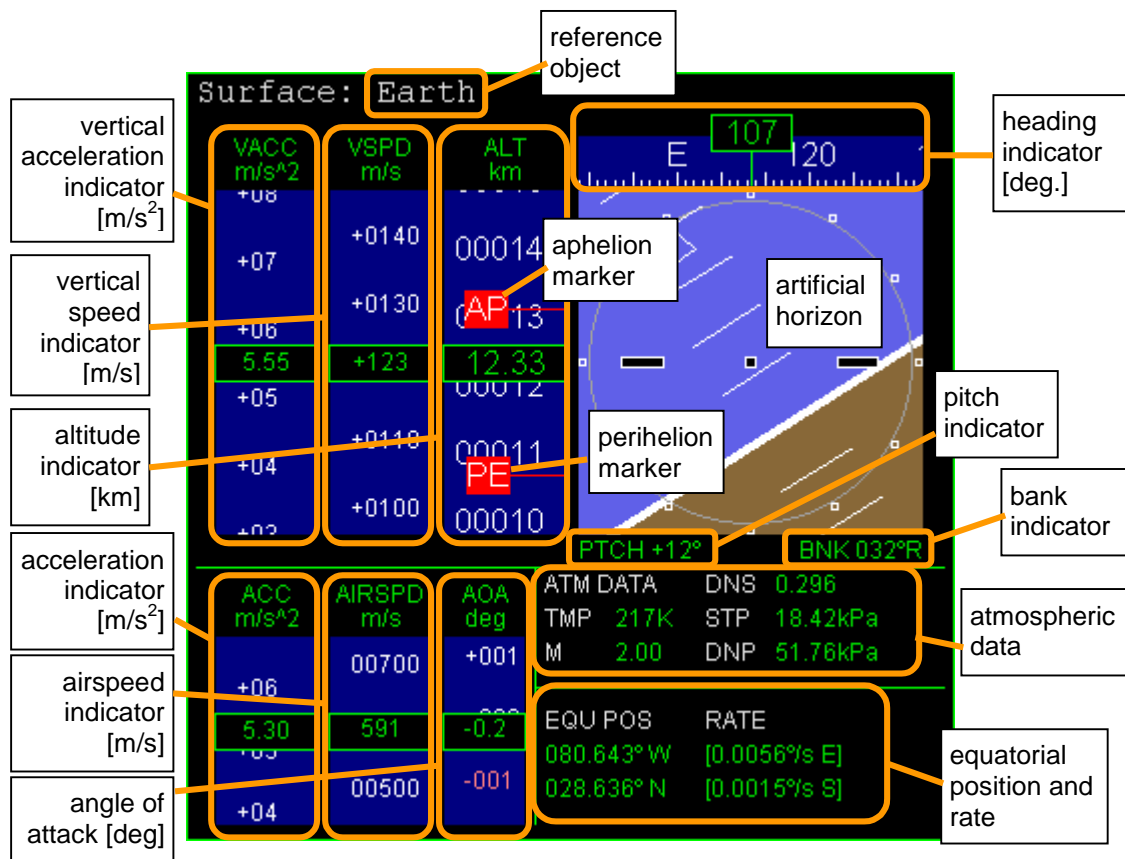


Figure 22: Surface MFD.

11.7 Map

The *Map* MFD mode shows a surface map of a planet or moon, including all available surface bases. The position of the currently selected base and the distance to it are displayed. In addition, the projections of the orbital planes of the ship and the selected orbital target onto the planet surface are plotted.

Key options:

	Open input box for reference planet/moon selection.
	Open a menu for target selection.



Figure 23: Map MFD.

Target base:

- **Pos:** Equatorial coordinates (longitude, latitude) of selected spaceport.
- **Dst:** Surface distance of ship's projected position to the target base.
- **Dir:** Direction of the target base as seen from the ship.

Target orbit:

- **Pos:** Equatorial coordinates (longitude, latitude) of the projected target position.
- **Alt:** Target altitude.



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.

11.8 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 14.3 on how to use this MFD mode.

Key options:

 	Input a new target object or target orbital parameters.
---	---

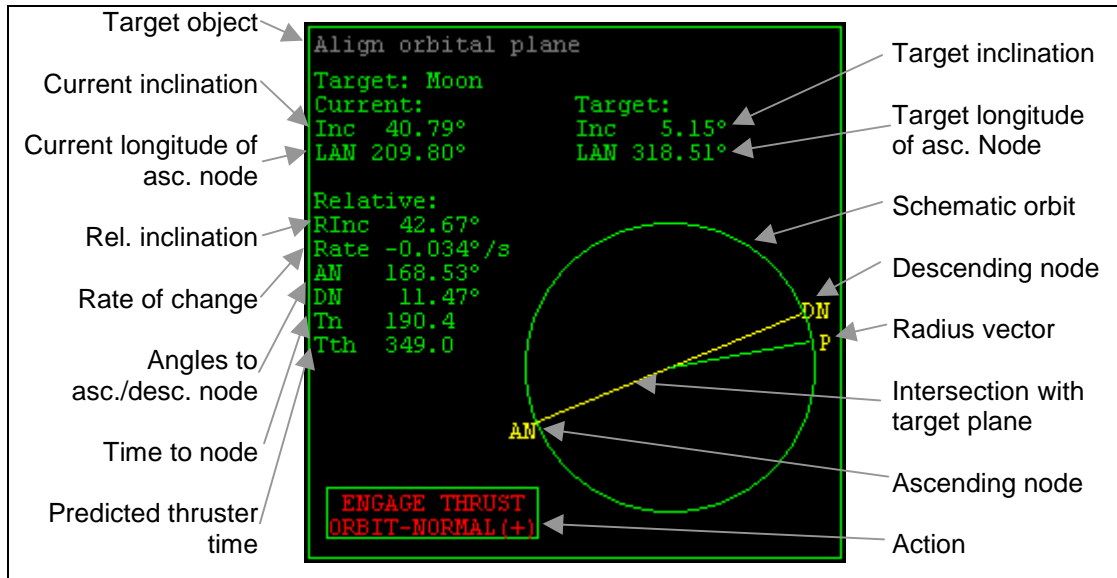


Figure 24: Align Orbital Plane MFD mode.

11.9 Synchronise orbit

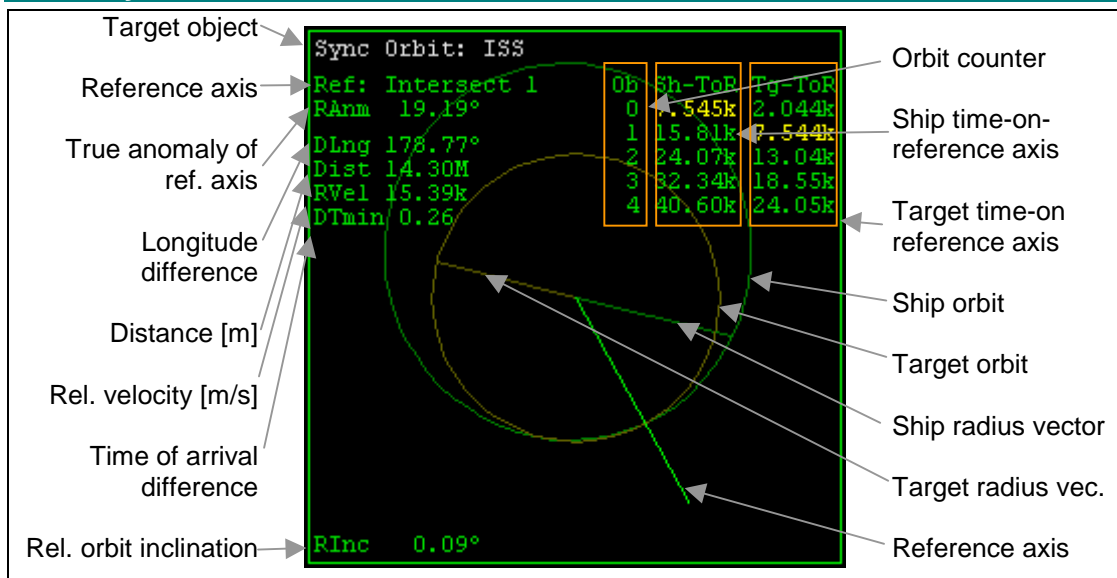


Figure 25: Synchronise Orbit MFD mode.

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

- **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 (RInc):** 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.

Key options:

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

For usage of this MFD mode in orbit synchronisation, see Section 14.5.

11.10 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).

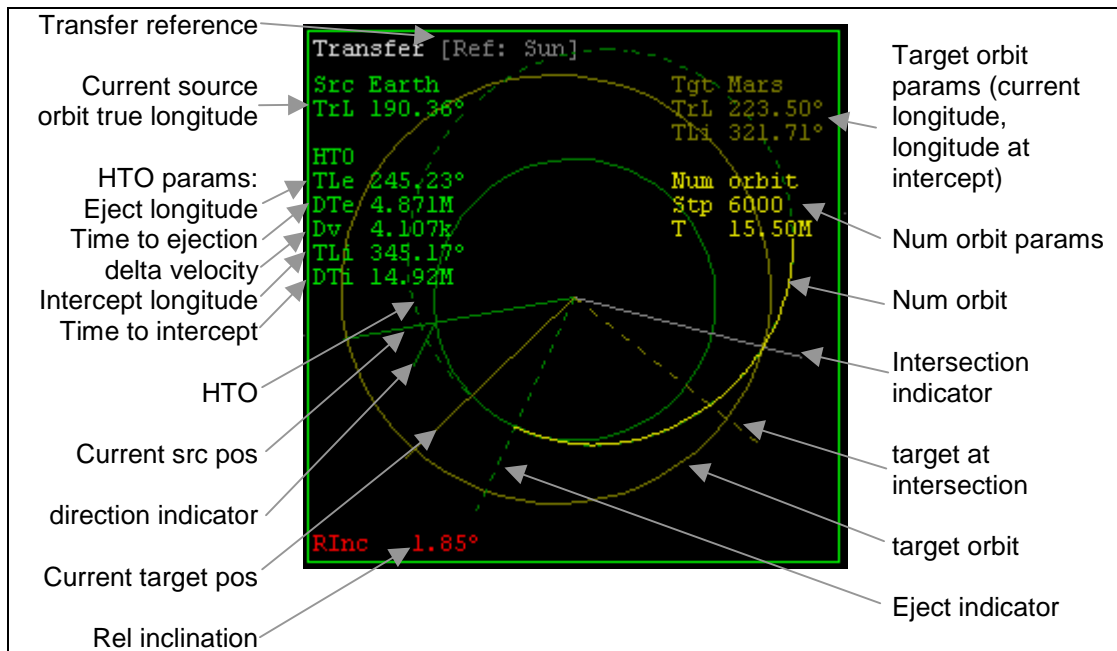


Figure 26: Transfer MFD mode.

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.

Key options:

	Open input box for reference planet/moon selection.
	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 multibody trajectory calculation.
	Refresh numerical trajectory, if displayed.
	Open input box for time step definition.
	Rotate transfer orbit ejection longitude.
	Decrease/increase transfer orbit major axis.

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][.]**/**[Shift][,]**) and the velocity change during the burn (adjusted with **[Shift][+]**/**[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 grey 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 grey 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 prograde (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 multibody 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 Δv can be estimated.

11.11 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 mode can be selected with **Shift**-**F1**-**Shift**-**P**.

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.

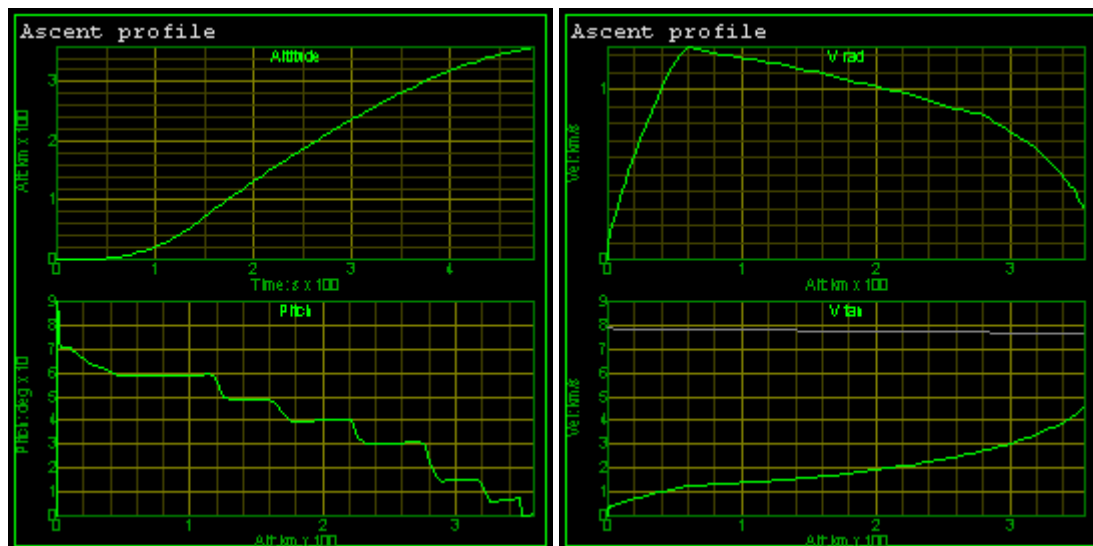


Figure 27: Ascent profile MFD mode, pages 1 and 2

Key options:

Shift - P	Switch display page.
Shift - A	Set altitude range.
Shift - R	Set radial velocity range.
Shift - 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 (V_{tan}), a grey 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.

12 Spacecraft controls

12.1 Main, retro and hover engines

Main thrusters accelerate the ship forward, *retro thrusters* accelerate it backward. Main and retro engines can be adjusted with **Ctrl** **+** **Numpad** (to increase main thrust or decrease retro thrust) and **Ctrl** **-** **Numpad** (to decrease main thrust or increase retro thrust). Main and retro thrusters can be killed with **Ctrl** ***** **Numpad**. The permanent setting can be temporarily overridden with **+** **Numpad** (set main thrusters to 100%) and **-** **Numpad** (set retro thrusters to 100%). If available, a joystick throttle control can be used to set main thrusters. The ship's acceleration **a** resulting from engaging main or retro thrusters depends on the force **F** produced by the engine and the ship's mass **m**:

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

In the absence of additional forces (such as gravitation or atmospheric drag) the spacecraft will move with constant velocity **v** as long as no engines are engaged. When engines are engaged, the ship's velocity will change according to

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

The current main/retro thruster setting and corresponding acceleration is displayed in the upper left corner of the HUD ("Main"). The indicator bar is green for positive (main) thrust, and yellow for negative (retro) thrust. The numerical acceleration value is in units of m/s². Note that for a fixed thruster setting **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** **Numpad** and decreased with **.** **Numpad**. 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.

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).

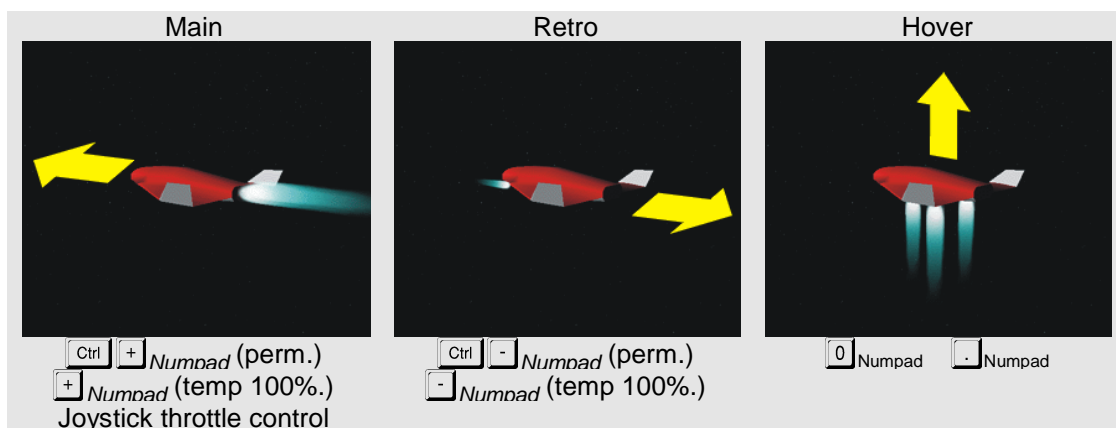


Figure 28: Acceleration from main, retro and hover thrusters.

The maximum thrust ratings for main, retro and hover thrusters as well as the current spacecraft mass are displayed in external spacecraft views. Values are in Newton (1N = 1kg m s⁻²).

12.2 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 **[7] Numpad**.

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

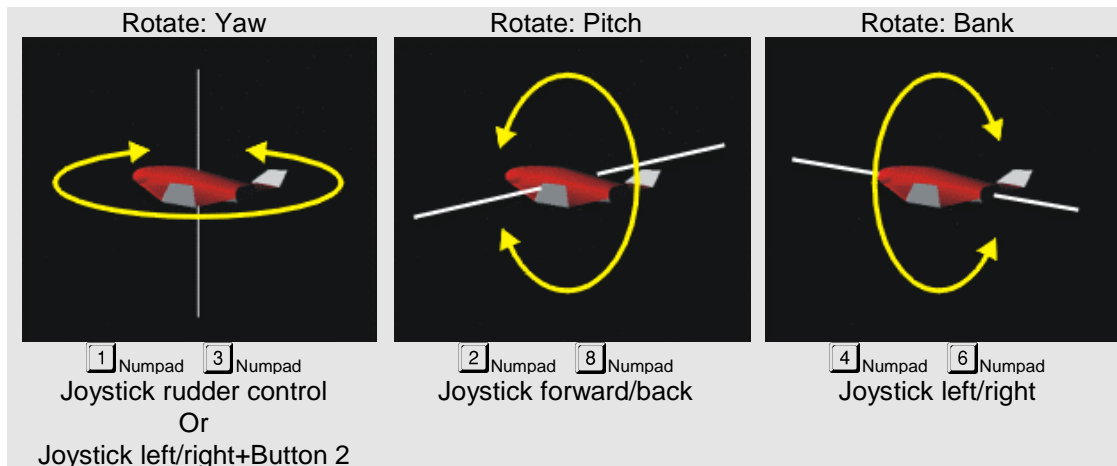


Figure 29: Attitude thrusters in rotational mode.

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

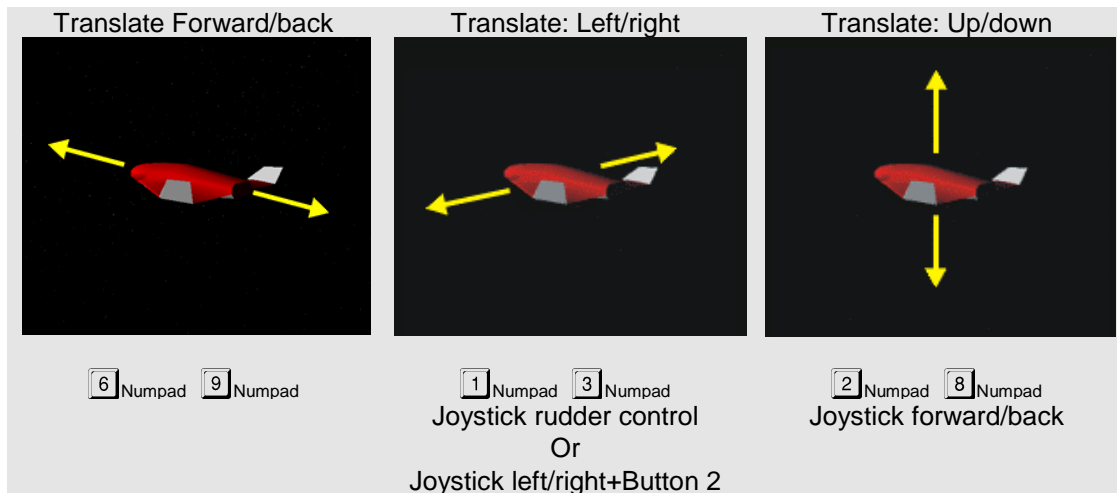


Figure 30: Attitude thrusters in translational (linear) mode.

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] Numpad**). This will automatically engage appropriate attitude thrusters to stop the ship's rotation. The kill rotation sequence only works in rotational attitude mode.

13 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 (**Shift** **C**), select a receiver (**Shift** **,** and **Shift** **.**), and scan the frequency (**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**).
- **NEW!** **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**).



Note: XPDR and IDS transmitters are only supported for vessel classes, not for dedicated space station classes used in previous Orbiter versions. To equip a space station with XPDR and IDS transmitters, it must be defined as a vessel.

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

14 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.

14.1 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.

14.2 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 (Shift-S) and Orbit (Shift-O) MFD modes.
- Engage hover thrusters to at least 10m/s^2 .
- Once free of the surface, turn towards east (90° on HUD compass ribbon).
- Raise nose to 70° 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° at 20km, 50° at 50km, 40° at 80km, etc).

- As the desired altitude is reached (e.g. 200km) 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.

14.3 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.

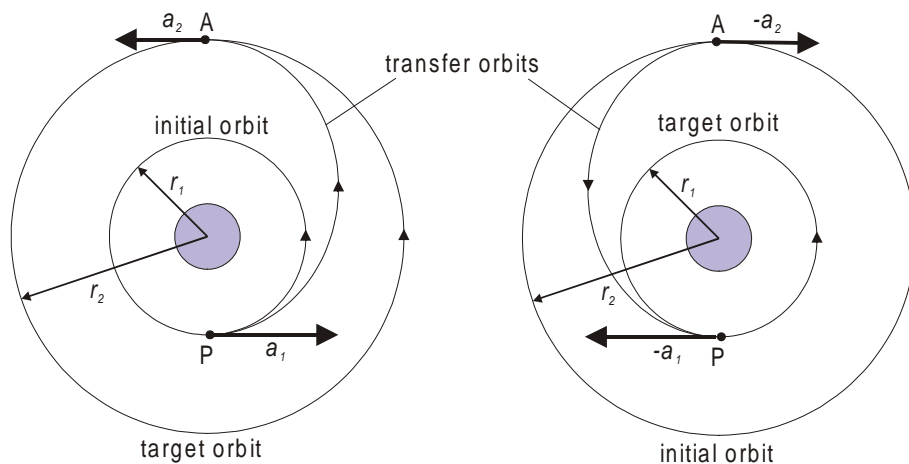


Figure 31: Moving into a higher orbit involves prograde acceleration at P and A (periapsis and apoapsis of the transfer orbit). Conversely, moving from the higher to the lower orbit requires retrograde acceleration at A and P.

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.

14.4 Rotating the orbital plane

In order to rendezvous with another orbiting body (e.g. a space station) or to prepare for transit to a moon or planet, the first step is to align the orbital plane (OP) of your ship with that of the target. Once you are in the same OP 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 (Ω).

The principal technique to rotate the OP is to point the spacecraft normal (perpendicular) to the current orbital plane, and to fire the engines. This will start to rotate the OP around an axis defined by your current radius vector. Therefore in order to align the orbit with a given target plane:

- Wait until you reach the intersection (node) of your orbit with the target plane
- Rotate the ship to point normal to the current orbit
- Fire the engines until the OP aligns with the target plane.

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 ΔT , thrusters should be engaged approximately $\frac{1}{2} \Delta T$ before intercepting the node.

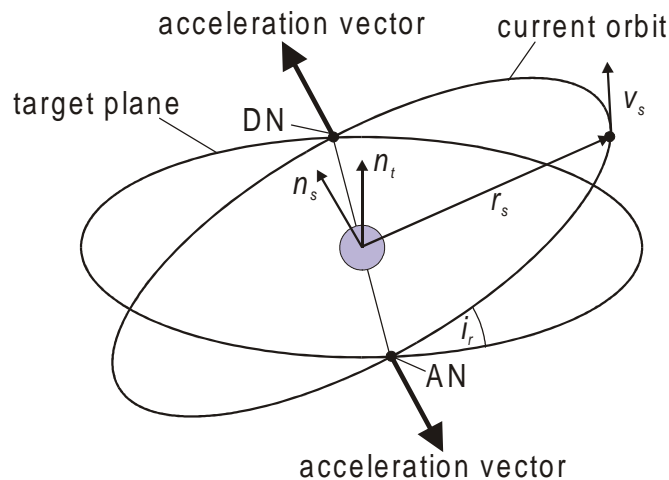


Figure 32: Alignment of the orbital plane. r_s : radius vector. v_s : velocity vector. AN: ascending node. DN: Descending node. n_s : normal of the current plane. n_t : normal of the target plane.

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 Figure 32).

In Practice:

- The *Align orbital plane* MFD mode (Shift-A, see Section 11.8) is designed to aid in plane alignment. Select the target object (Shift-T).
- The HUD should be in Orbit mode. As your ship approaches the intersection with the target plane, rotate it normal (if at DN) or anti-normal (if at AN) to the current orbital plane. Use the HUD Orbit inclination ladder for this.
- As soon as the time to node (T_n) reaches half the estimated burn time (T_{th}) 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.
- 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 manoeuvre make sure your orbit does not become unstable. Watch in particular for the eccentricity (use the Orbit MFD to monitor this).

14.5 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 manoeuvre 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* (Shift-Y) 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 14.3), 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.

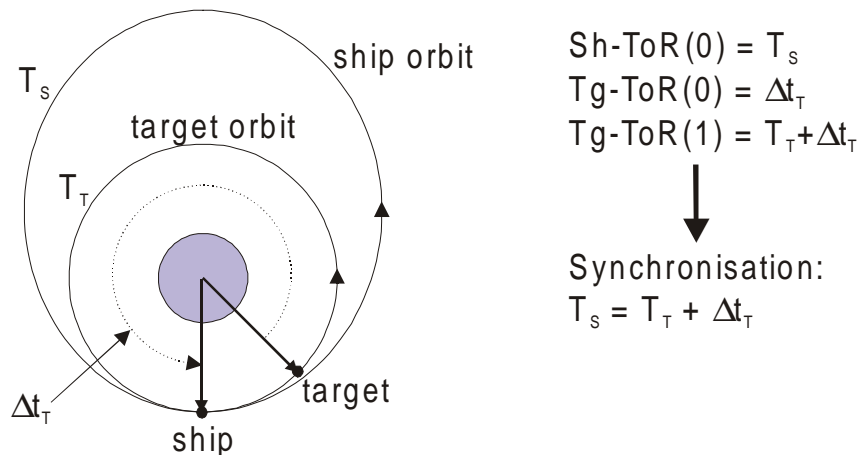


Figure 33: A transition orbit to intercept the target at the next periapsis passage.

Notes:

- Instead of increasing the apoapsis distance one could fire retrograde and reduce the periapsis distance in this manoeuvre. 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 manoeuvre. 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.

14.6 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° (about 6 times as steep as standard aircraft approach slopes!). The VASI is set up for a 1.5° slope during the final flare up prior to touchdown.

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.

	Above glide slope
	Slightly above glide slope
	On glide slope
	Slightly below glide slope
	Below glide slope

Figure 34: 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.

	Above glide slope
	On glide slope
	Below glide slope

Figure 35: VASI indicator signals

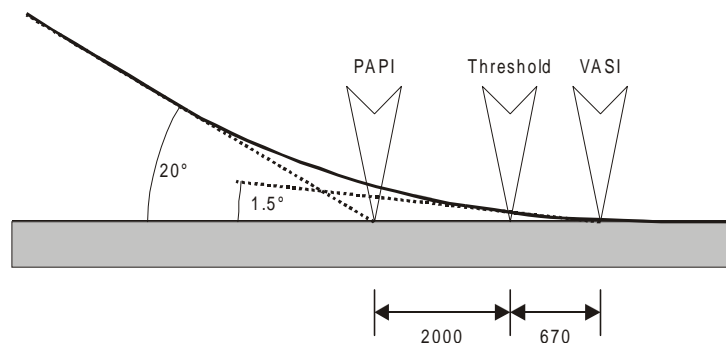


Figure 36: SLF Shuttle approach path

14.7 Docking

Note: the procedure below refers to docking at a space station which is defined as a vessel. Docking at dedicated station classes used in previous Orbiter versions requires some modifications.

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

- Turn on the *Docking* MFD (**Shift** **D**) 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 (\oplus) and fire main thrusters until velocity value (V) approaches zero.
- Rotate the ship to face the station (\ominus 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 manoeuvre.
- To disengage from the docking port, press **Ctrl** **D**.

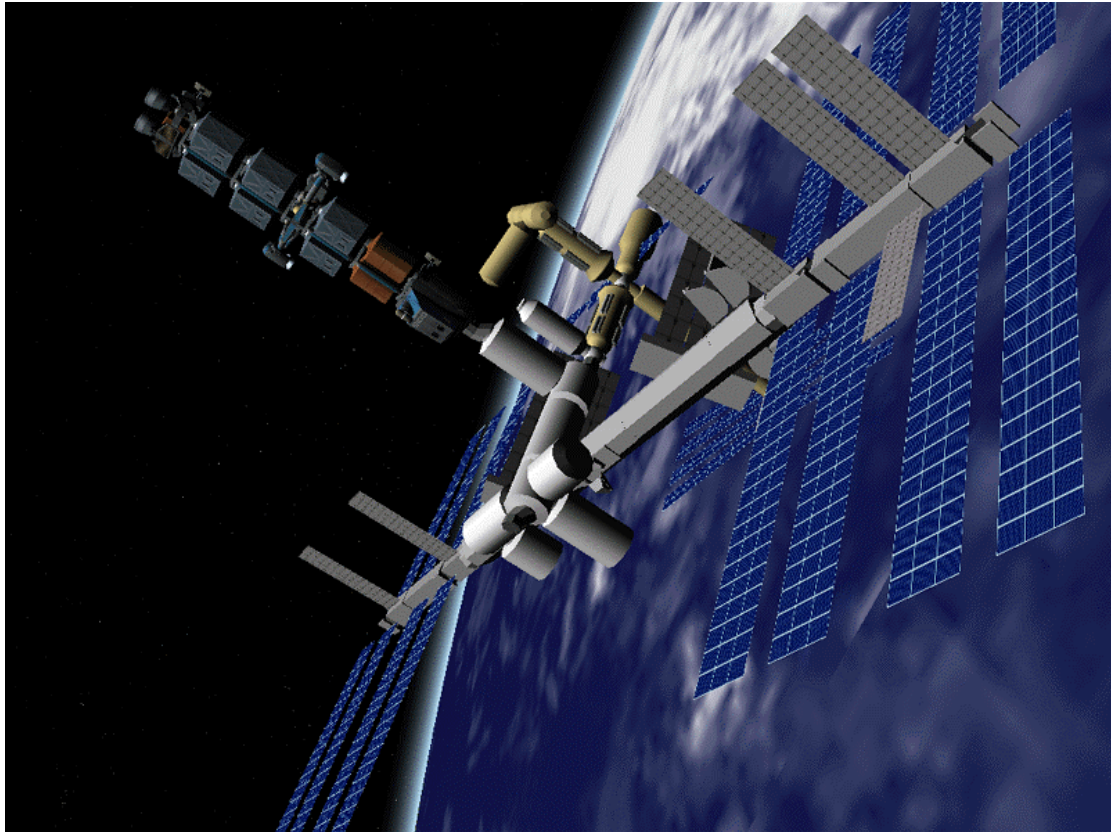


Figure 37: A Shuttle-A class cargo ship after successfully completed docking approach to the ISS.

Notes:

- For precise attitude control with the keyboard use the attitude thrusters in “low power” mode (Ctrl + Numpad key).
- You can only dock to a port you have been cleared for.
- 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 Numpad5 (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.

15 Extra functionality

NEW!

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.

Many more plugins are available from 3rd 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.

Some plugins are accessible via the Custom functions dialog. Press **Ctrl F4** to get a list of the available functions.

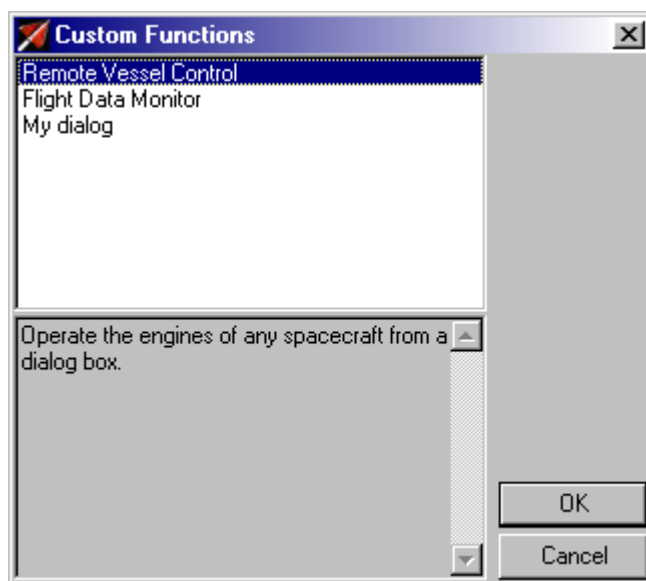


Figure 38: Custom functions dialog for access to plugin functions.

15.1 Frame rate monitor

NEW!

This is a little dialog box to keep track of Orbiter's frame rate performance. It shows the frames per second (FPS) 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. This function is only available if the *Framerate* module is active (see Section 3.4), and is accessible via the *Frame Rate* entry in the *Custom functions* panel (**Ctrl F4**).

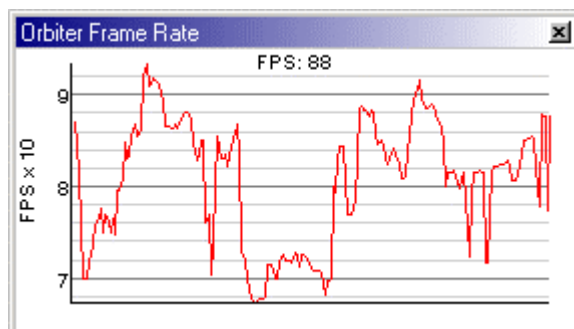


Figure 39: Frame rate monitor.

15.2 Remote vessel control

NEW!

The *Remote Vessel Control* plugin allows to remotely control the engines of all spacecraft. This tool is available only if the *Rcontrol* module is active. The remote control dialog box is accessible via the *Custom functions* panel (**Ctrl** **F4**). 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.

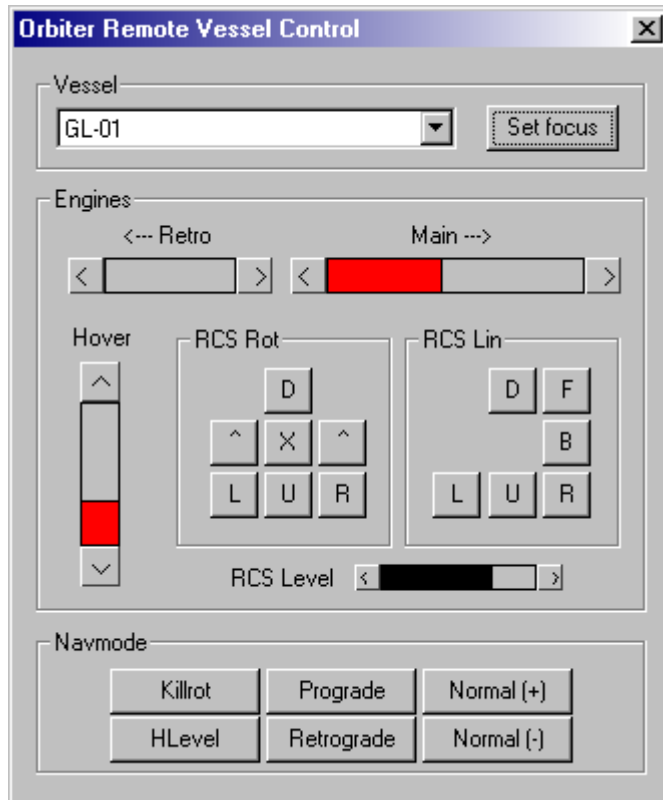


Figure 40: Remote vessel control dialog.

15.3 Flight data monitor

NEW!

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

The following parameters are currently supported:

- Altitude against time
- Airspeed against time
- Lift and drag force against time
- Lift over drag ratio (L/D) against time
- Mach number against time
- Static and dynamic pressure against time

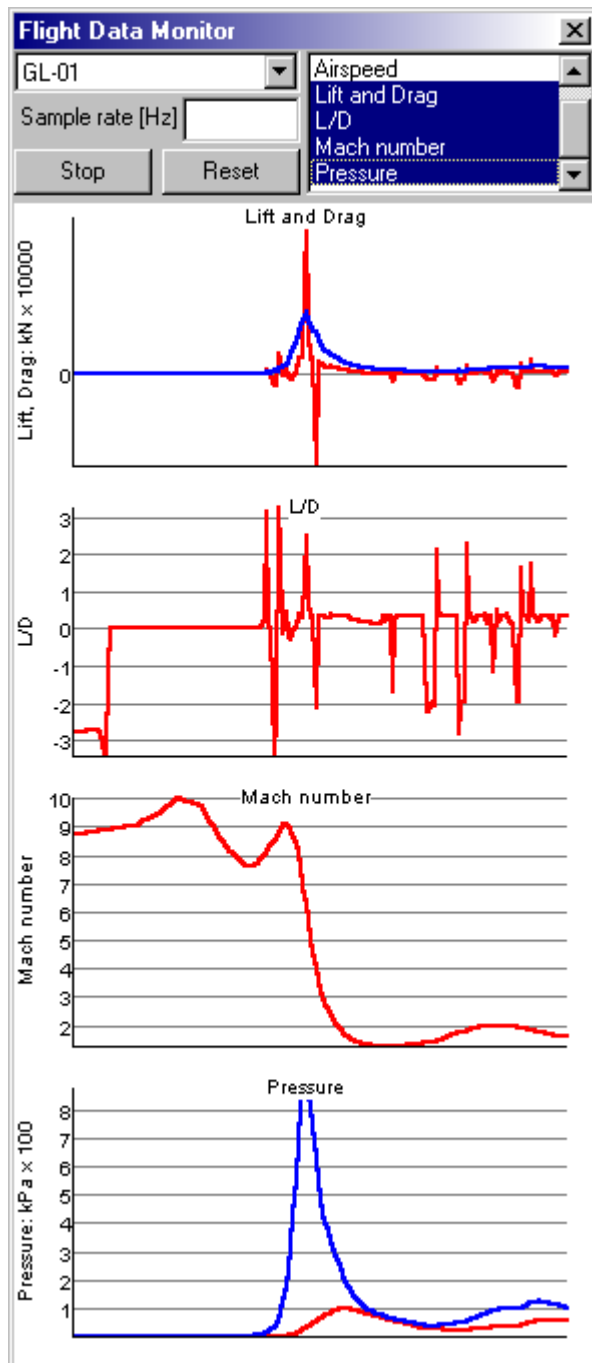


Figure 41: Flight data monitor

16 Flight checklists

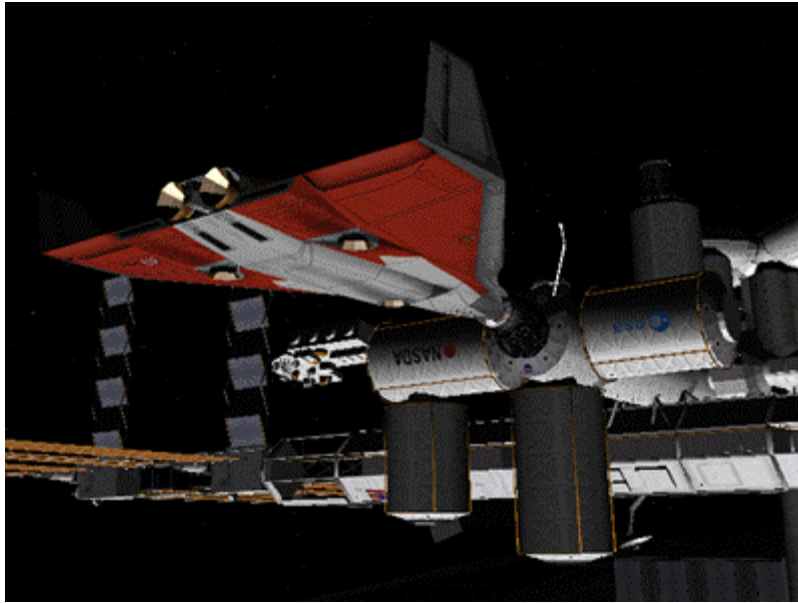
This section contains point-by-point checklists for some complete flights. While flying these checklists, you may want to save regularly (**Ctrl** **S**), so you can pick up from a previous state if necessary.

16.1 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 manoeuvre 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 (**↑** *Cursorpad*) 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 8.1. For details on MFD modes, see section 11.
- 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 **T**. (Each time you press **T**, time accelerates by a factor of 10). As you approach launch time, switch back to real-time by pressing **R** until the "WRP" indicator in the top left corner of the screen disappears.
- Engage main engines (**Ctrl** **+** *Numpad*) 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 **2** *Numpad*) to rotate.
- Climb at 10° and retract the landing gear (**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 left side of the instrument panel) or by pressing **Ctrl** **/** *Numpad*. 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 (the "⊕" symbol on the HUD) should stay above 0°.
- Switch the right MFD to Orbit mode (**Shift** **O**). 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°. You will see how your orbit trajectory (green curve in the Orbit MFD) grows.
- Cut thrusters when your apoapsis distance (highest point of the orbit) reaches 6.731M (the "ApD" entry in the left column of the Orbit MFD). This corresponds to an altitude of 360 km.
- To enter a circular orbit, you need to perform a further burn at apoapsis. The apoapsis point should be on the opposite point of your orbit and will take about 45 minutes to reach, so you probably want to time-accelerate. (Check the "ApT" entry in the Orbit MFD – this shows the remaining time to reach apoapsis). At apoapsis, press the "Prograde" button to turn prograde. Once the velocity marker ("⊕") is centered on the screen, engage main thrusters until orbit eccentricity ("Ecc") reaches a minimum, and periapsis distance ("PeD") equals ApD. (this will require only a short burn!)
- Switch to Orbit HUD mode (**H**, **H**)

- Switch the left MFD to Align Orbital Plane (**Shift** **A**). Select ISS (**Shift** **T**, 'ISS').
- Ideally the orbital planes should already be roughly aligned (RInc within 5°). 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° 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°) 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 (**Shift** **Y**). 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 0) 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(0) 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 now until you reach the reference point.
- Tune your NAV receivers to the station's navaid radio transmitters: Select Comm MFD mode (**Shift** **C**), and tune NAV1 to 131.30 kHz (ISS XPDR frequency) and NAV2 to 137.40 kHz (Dock 1 IDS frequency).
- Switch to Docking HUD mode (**H**, **H**) and to Docking MFD (**Shift** **D**).
- 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 (translational) mode for this. Switch between linear and rotational mode with the Numpad/ 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 "Λ" 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.

16.2 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 MFD: Left-**[Shift]** **[A]**, 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" (T_n) value in the Align plane MFD is down to 500.
- Prepare attitude for the burn: click the "Orbit normal (+)" button.
- 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 (click "Orbit Normal (-)")
- 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 manoeuvre is complete, switch the HUD to Docking mode (**[H]**, **[H]**), and switch one of the MFD displays to Docking (**[Shift]** **[D]**) Slave both HUD and MFD to NAV1.
- Proceed with the docking manoeuvre 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.

16.3 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 MIR. You are currently over the Pacific, already at the correct location for the deorbit burn.
- Undock (**Ctrl** **D**), and engage retros for a few seconds (**-** *Numpad*) to get clear of the station.
- Close the nose cone (**K**).
- Turn retrograde (**I**).
- 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 perihel distance (PeD in Orbit MFD) has decreased to 5.600M.
- Turn prograde (**I**).
- When attitude has stabilised, roll the glider level with the horizon (**L**).
- Switch to Surface HUD mode (**H**).
- Turn left MFD into Surface mode (Left-**Shift** **S**).
- 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** **/** *Numpad*), 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 MFD mode (Right-**Shift** **C**).
- Turn the right MFD to Horizontal Situation Indicator (HSI) mode (Right-**Shift** **H**). 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.

17 ORBITER configuration

Configuration files allow the customisation of various aspects of ORBITER. Configuration files have file extension .cfg. The format is

item = value

A semicolon starts a comment, continuing to the end of line.

All configuration files except for the master file (see below) are located in a subdirectory defined by the ConfigDir entry in the master file, usually “.\Config”.

To find out how to set up the configuration file of a *spacecraft class*, see the separate *3DModel* documentation, contained in the Orbitersdk\doc folder (you need to install the SDK package to get this document). This document also contains information on the mesh file format for the vessel's 3D model, and other hints on how to create your own spacecraft.

Scenarios (simulation startup definitions) are located in a subdirectory defined by the ScenarioDir entry in the master file, usually “.\Scenarios”. They have file extension .scn.

17.1 Master configuration file

The master configuration file *Orbiter.cfg* is located in the Orbiter main directory. It contains general settings for graphics modes, subdirectory locations, simulation parameters, etc. Note that manual editing of Orbiter.cfg should no longer be necessary, because most parameters can be accessed from the Launchpad dialog.

Item	Type	Description
ConfigDir	String	Subdirectory for configuration files (default .\Config\)
MeshDir	String	Subdirectory for mesh files (default .\Meshes\)
TextureDir	String	Subdirectory for textures (default .\Textures\)
HightexDir	String	Subdirectory for alternative high-resolution planetary textures (default: .\Textures2\)
ScenarioDir	String	Subdirectory for scenarios (default: .\Scenarios\)
DeviceIndex	Int	Enumeration index for current 3D device (<i>do not edit manually</i>)
DeviceForceEnum	Bool	If TRUE, enumerate 3D devices at each start
ModelIndex	Int	Screen mode index (<i>do not edit manually</i>)
Fullscreen	Bool	TRUE for fullscreen mode, FALSE for windowed mode
Stereo	Bool	<i>Currently not used.</i>
WindowWidth	Int	Horizontal window size for windowed modes.
WindowHeight	Int	Vertical window size for windowed modes. The ratio WindowWidth/WindowHeight should be approximately 4/3.
RGBDepth	Int	Screen colour depth (BPP) for fullscreen modes.
JoystickIndex	Int	Enumeration index for current joystick (0=none; <i>do not edit manually</i>)
JoystickThrottle Saturation	Int	Saturation zone for joystick throttle control (0-10000). A setting of 9000 means that the throttle will saturate over the last 10% of its range at either end. Default: 9500
JoystickDeadzone		Deadzone at joystick axis centres (0-10000). A setting of 2000 means the joystick is considered neutral within 20% from the central position. Default: 2500
AmbientLevel	Int	Ambient light level (brightness of not directly lit surfaces). Valid range is 0-15.
NumStar	Int	Number of background stars (<= 15984). Default: 3000
StarBrightness	Float	Brightness scaling factor for background stars (0.2 ...2). Default: 1.0
ConstellationCol	RGB	Colour for constellation lines. Default: 0.3 0.3 0.3
UnlimitedFuel	Bool	Ignore spacecraft fuel consumption. Default: FALSE
FlightModel	Int	Flight model realism level (currently supported: 0 and 1)

MFDTransparent	Bool	Make onscreen multifunctional displays transparent. Default: TRUE
EnableShadows	Bool	Enable/disable object shadows on planet surfaces.
EnableSunGlare	Bool	Enable sun glare effects.
EnableClouds	Bool	Enable rendering of planetary cloud layers.
EnableNightlights	Bool	Enable rendering of night lighting effects of planetary surfaces.
NightlightBrightness	Int	Brightness level of night lighting effects (0-255)
EnableWaterReflection	Bool	Enable rendering of specular reflections from oceanic surfaces.
EnableHorizonHaze	Bool	Enable rendering of atmospheric effects at the horizon.
EnableSpecularReflection	Bool	Enable specular reflection effects from polished surfaces.
InstrumentUpdateInterval	Float	Interval between MFD display updates (seconds)
PanelScale	Float	Scaling factor for instrument panel display.
PanelScrollSpeed	Float	Speed factor for panel scrolling (pixels per second)
PlanetPatchRes	Float	Resolution factor for planet surfaces. Valid range is 0.1 to 10. Higher values produce higher resolution planetary surfaces at a given apparent radius, but reduce performance.
DialogFont_Scale	Float	Scaling factor for dialog font size. Default: 1.0
DialogFont1_Face	String	Standard dialog font face. Default: Arial
ActiveModules	List	List of active plugin modules

17.2 Planetary systems

Planetary systems contain stars, planets and moons. Each planetary system requires at least one star.

Star entries:

Star<i> = Name

where <i> is an index running from 1 upward. (*note: planetary systems with more than one central star are not currently supported*).

Planet entries:

Planet<i> = Name

where <i> is an index running from 1 upward.

Moon entries:

<Planet>:Moon<i> = Name

where <Planet> is the name of a planet defined before, and <i> is an index enumerating the moons of this planet, running from 1 upward.

Example:

Star1 = Sun
Planet1 = Mercury
Planet2 = Venus
Planet3 = Earth
Earth:Moon1 = Moon
Planet4 = Mars
Mars:Moon1 = Phobos
Mars:Moon2 = Deimos

17.3 Planets

Planet configuration files define the planet's orbital, physical and visual parameters. For an example see Config\Earth.cfg.

1. General parameters

Item	Type	Description
Name	String	Planet name
Module	String	Name of dynamic link library performing calculations for the planet (default: none)
ErrorLimit	Float	Max. rel. error for position/velocity calculations (only used if the module supports precision adjustment)
EllipticOrbit	Bool	If TRUE, use analytic 2-body solution for planet position/velocity calculation, otherwise update dynamically (ignored if module supports position/velocity calculation)
HasElements	Bool	If TRUE, the initial position/velocity is calculated from the provided set of orbital elements, otherwise from an explicit position/velocity pair (ignored if module supports position/velocity calculation)

Notes:

- If the module calculates the planet position and velocity from perturbation terms, then the value of ErrorLimit will affect the number of terms used for the calculation. A lower value will increase the number of required terms, and thus the calculation time. The valid range for ErrorLimit depends on the module, but is typically $1e-3 \leq \text{ErrorLimit} \leq 1e-8$.

2. Orbital parameters (Ignored if module supports position/velocity calculation or HasElements = FALSE)

Item	Type	Description
Epoch	Float	Orbital element reference epoch (e.g. 2000)
EIReference	Flag	ParentEquator or Ecliptic: orbit reference frame (default: Ecliptic)
SemiMajorAxis	Float	Orbit semi-major axis [m]
Eccentricity	Float	Orbit eccentricity
Inclination	Float	Orbit inclination against reference plane [rad]
LongAscNode	Float	Longitude of ascending node [rad]
LongPerihelion	Float	Longitude of periapsis [rad]
MeanLongitude	Float	Mean longitude at epoch [rad]

3. Physical parameters

Item	Type	Description
Mass	Float	Planet mass [kg]
Size	Float	Mean planet radius [m]

4. Rotation elements

Item	Type	Description
SidRotPeriod	Float	Siderial rotation period [s]
SidRotOffset	Float	Rotation at epoch [rad]
Obliquity	Float	Obliquity: angle between rotation axis and normal of reference plane at epoch [rad]
LAN	Float	Longitude of projection of rotation axis onto reference plane [rad]

5. Atmospheric parameters (only required if planet has atmosphere)

Item	Type	Description
AtmPressure0	Float	(Mean) atmospheric pressure at zero altitude [Pa]
AtmDensity0	Float	(Mean) atmospheric density at zero altitude [kg/m ³]
AtmColor0	Vec ₃	RGB triplet for atmospheric colour at ground level (0-1 each)
AtmAltLimit	Float	altitude limit beyond which atmospheric effects can be ignored [m]
AtmHorizonAlt	Float	altitude scale for horizon haze rendering [m]. Default: 0.01 of planet radius.

6. Cloud parameters (only required if planet contains a cloud layer)

Item	Type	Description
CloudAlt	Float	Altitude of cloud layer [m]
CloudRotPeriod	Float	Rotation period of cloud layer against surface [s] (default: 0 – static cloud layer)

7. Visualisation parameters

Item	Type	Description
MaxPatchResolution	Int	Max. resolution level for surface texture maps (1 ... 8)
MinCloudResolution	Int	Min. resolution at which clouds are rendered as separate layer (1 ... MaxPatchResolution)
MaxCloudResolution	Int	Max. cloud resolution level (MinCloudResolution ... 8)
MaxDynamicResolution	Int	Reserved

8. Surface bases (only required if planet contains surface bases)

Item	Type	Description
NumBases	Int	Number of spaceports on planet surface
Base<X>	List	<X> = 1 ... NumBases. List contains: Name, longitude [deg], latitude [deg]

9. Ground-based observer sites

This list contains the locations for ground-based observers (launch cameras, spectators, etc.) which can be selected in the Camera dialog. The format of the list is

```
BEGIN_OBSERVER
  <observer list>
END_OBSERVER
```

List entries have the following format:

<site>:<spot>: <lng> <lat> <alt>

where

<site> a name which identifies the site (e.g. KSC)
<spot> the particular location at the site (e.g. Launch pad 39)
<lng> <lat> observer position (equatorial coordinates) [deg]
<alt> observer altitude [m] (>0)

The easiest way to find the coordinates for a new observer spot is to open the *Camera* dialog (**Ctrl** **F1**), and select a nearby location under the *Ground* tab. Then move the camera to the new spot using **Ctrl** **↓** **↑** **→** **←** and **Page Up** **Page Down**. The coordinates are displayed in the dialog and can be directly copied into the configuration file.

10. Navbeacon transmitter list

This list contains all navigation radio transmitter specs except those directly associated with a spaceport (see section 17.4). The list format is as follows:

```
BEGIN_NAVBEACON
  <NAV list>
END_NAVBEACON
```

List entries have the following format:

<type> <id> <lng> <lat> <freq> [<range>]

where

<type> transmitter type. currently supported: VOR
<id> identifier code (up to 4 letters)
<lng> <lat> transmitter position (equatorial coordinates) [deg]
<freq> transmitter frequency [kHz]

<range> transmitter range [m] (default: 500 km)

To implement a custom DLL module for planet position/velocity calculations, see SDK documentation.

To add a new planet to a planetary system the following steps are required:

1. Add an entry for the planet in the planetary system configuration file (see previous section):
Planet<X> = <Planetname>
2. Create a configuration file *<Planetname>.cfg* for the new planet in the “Config” subdirectory, with entries as listed above.
3. Create the required surface texture maps up to the specified resolution (see Section 0).
4. Optionally, create a monochrome (green on black) surface outline bitmap (256x128, BMP) to be used by the Map MFD. The file name should be *<Planetname>M.bmp*.

17.4 Surface bases

To create a new spaceport on the surface of a planet the following steps are required:

1. In the planet’s configuration file
 - Increase the entry for *NumBases* by 1, or add the line
NumBases = 1
if no bases are defined for the planet yet.
 - Add a line
Base<X> = <Name> <Long> <Lat>
where *<X>* is the base’s numeric ID (should be set to the new value of *NumBases*), *<Name>* is the base’s name (which identifies its configuration file), and *<Long>* and *<Lat>* define the base’s position on the planet surface in equatorial coordinates (degrees). Eastern Longitudes and northern Latitudes are counted positive, western longitudes and southern latitudes are negative.

Example:

```
NumBases = 1
Base1 = Habana -82.5 +23.0
```

2. Create a configuration file for the base, *<Name>.cfg*, where *<Name>* corresponds to the base’s name in the planet configuration file. The format of the base definition file is as follows:

```
NAME = <Base name>
SIZE = <size>
OBJECTSIZE = <osize>
MAPOBJECTSTOSPHERE = <spmap>

BEGIN_NAVBEACON
  <NAV list>
END_NAVBEACON

BEGIN_OBJECTLIST
  <Object list>
END_OBJECTLIST
```

where *<Base name>* is the base’s (logical) name which need not correspond to the entry in the planet configuration file. *<Size>* is the base’s overall radius in meters.

<osize> is the size of a “typical” object (building, etc.) This value is used by Orbiter to determine up to what camera distance base objects will be rendered. Objects will not be

rendered if the apparent size of an object of size *<osize>*, located at the centre of the base, would be smaller than 1 pixel. The default value for *<osize>* is 100.0.

<spmap> is a boolean (default: false).

If *true*, the objects in the object list will be automatically adjusted in elevation to correct for the planets curvature. This means that objects with elevation 0 will be mapped onto altitude 0 of the planet surface.

If *false*, elevation 0 maps onto the flat horizon plane of the base reference point.

Note: Currently this function is only implemented for a limited number of base object types.

<NAV list> contains a list navigation radio transmitters associated with the base. The format is identical to that of the Planet config file (see section 17.3).

<Object list> contains a list of objects which make up the visual elements of the base. See next section for details.

17.5 Adding objects to surface bases

Surface bases are composed of objects (buildings, train lines, hangars, launch pads, etc.) The configuration file for each surface base contains a list of its objects:

```
BEGIN_OBJECTLIST
  <Object 0>
  <Object 1>
  ...
  <Object n-1>
END_OBJECTLIST
```

Each object entry in the list defines a particular object and its properties (type, position, size, textures, etc.). An object can either be a pre-defined type or a generic mesh. Each object entry has the following format:

```
<Type>
  <Parameters>
END
```



Note that textures used by base objects must be listed in the texture list of the *Base.cfg* configuration file.

The following pre-defined object types are currently supported:

BLOCK

A 5-sided “brick” (without a floor) which can be used as a simple generic building, or as part of a more complex structure. The following parameters are supported:

Parameter	Type	Description
POS	V	Centre of the block's base rectangle (in local coordinates of the surface base). Note that the y-coordinate is the elevation above ground. Default: 0 0 0
SCALE	V	Object size in the three coordinate axes. Default: 1 1 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX1	S F F	Texture name and u,v scaling factors for walls along the x-axis. Default: none
TEX2	S F F	Texture name and u,v scaling factors for walls along the z-axis. Default: none
TEX3	S F F	Texture name and u,v scaling factors for roof. Default: none

(V=Vector, F=Float, S=String)

HANGAR

A hangar-type building with a barrel-shaped roof. The following parameters are supported:

Parameter	Type	Description
POS	V	Centre of the object's base rectangle (in local coordinates of the surface base). Note that the y-coordinate is the elevation above ground. Default: 0 0 0
SCALE	V	Object size in the three coordinate axes. Default: 1 1 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX1	S F F	Texture name and u,v scaling factors for walls. Default: none
TEX2	S F F	Texture name and u,v scaling factors for front gate. Default: none
TEX3	S F F	Texture name and u,v scaling factors for roof. Default: none

(V=Vector, F=Float, S=String)

HANGAR2

A hangar-type building with a tent-shaped roof. The following parameters are supported:

Parameter	Type	Description
POS	V	Centre of the object's base rectangle (in local coordinates of the surface base). Note that the y-coordinate is the elevation above ground. Default: 0 0 0
SCALE	V	Object size in the three coordinate axes. Default: 1 1 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX1	S F F	Texture name and u,v scaling factors for front and back walls. Default: none
TEX2	S F F	Texture name and u,v scaling factors for side walls. Default: none
TEX3	S F F	Texture name and u,v scaling factors for roof. Default: none
ROOFH	F	Roof height from base to ridge. Default: ½ building height.

(V=Vector, F=Float, S=String)

TANK

A fuel tank-like upright cylinder with flat top. The following parameters are supported:

Parameter	Type	Description
POS	V	Centre of the object's base circle (in local coordinates of the surface base). Note that the y-coordinate is the elevation above ground. Default: 0 0 0
SCALE	V	Cylinder radii in x and z, and height in y. Default: 1 1 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
NSTEP	I	Number of segments to approximate circle. Default: 12
TEX1	S F F	Texture name and u,v scaling factors for mantle. Default: none
TEX2	S F F	Texture name and u,v scaling factors for top.

(V=Vector, F=Float, I=Integer, S=String)

RUNWAY

Texturing for a runway. The texture mapping can be split into segments, to allow inclusion of markings, overruns, etc. This does not include any lighting (see RUNWAYLIGHTS).

Parameter	Type	Description
END1	V	First end point of runway (center line), including any overruns to be textured.
END2	V	Secon end point of runway (center line).
WIDTH	F	Runway width [m]
ILS1	F	Localiser frequency for approach towards END1 (108.00 to 139.95). Default: No ILS support
ILS2	F	Localiser frequency for approach towards END2 (108.00 to 139.95). This can be the same frequency as ILS1. Default: No ILS support.
NRWSEG	I	Number of texture segments

RWSEGx	I F F F F F	Definition of segment x (x = 1...NRWSEG). Parameters: 1. Number of mesh sub-segments (≥ 1) 2. Fractional length of segment (sum of all segments must be 1) 3. texture coordinate u_0 of segment 4. texture coordinate u_1 5. texture coordinate v_0 6. texture coordinate v_1
RWTEX	S	Texture name for all segments

(V=Vector, F=Float, I=Integer, S=String)

RUNWAYLIGHTS

Complete lighting for a single runway, including optional Precision Approach Path Indicator (PAPI) and Visual Approach Slope Indicator (VASI) – see section 14.6. Runway markers are turned off during daytime, but PAPI and VASI indicators are always active.

Parameter	Type	Description
END1	V	First end point of runway (center line).
END2	V	Second end point of runway (center line).
WIDTH	F	Runway width [m]
COUNT1	I	Number of lights along the runway center line (≥ 2). Default: 40
PAPI	F F F	Precision Approach Path Indicator (PAPI). Default: no PAPI. Parameters: 1. Designated approach angle [deg] 2. Approach cone aperture [deg] 3. Offset of PAPI location from runway endpoints. [m]
VASI	F F F	Visual Approach Slope Indicator (VASI). Default: no VASI Parameters: 1. Designated approach angle [deg] 2. Distance between white and red indicator lights [m] 3. Offset of VASI (red bar) location from runway endpoints [m]

(V=Vector, F=Float, I=Integer)

BEACONARRAY

A linear array of illuminated beacons, usable e.g. for taxiway night lighting.

Parameter	Type	Description
END1	V	First end point of beacon array (in local coordinates of the surface base). Note that the y-coordinate is the elevation above ground.
END2	V	Second end point of beacon array
COUNT	I	Number of beacons in the array (≥ 2). Default: 10
SIZE	F	Size (radius) of each beacon light. Default: 1.0
COL	F F F	Beacon colour (RGB) Valid range: 0..1 for each value. Default: 1 1 1 (white)

(V=Vector, F=Float, I=Integer)

SOLARPLANT

A grid of ground-mounted solar panels, smart enough to align themselves with the Sun. The following parameters are supported:

Parameter	Type	Description
POS	V	Centre position of the panel grid. Default: 0 0 0
SCALE	F	Scaling factor for each panel. Default: 1
SPACING	F F	Distance between panels in x and z direction. Default: 40 40
GRID	I I	Grid dimensions in x and z direction. Default: 2 2
ROT	F	Rotation of plant around vertical axis (degrees). Default: 0
TEX	S [F F]	Texture name and u,v scaling factors for panels. Default: none

(V=Vector, F=Float, I=Integer, S=String)

TRAIN1

A monorail-type train on a straight track. The following parameters are supported:

Parameter	Type	Description
END1	V	First end point of track
END2	V	Second end point of track
MAXSPEED	F	Maximum speed of train [m/s] Default: 30
SLOWZONE	F	Distance over which train slows down at end of track [m] Default: 100
TEX	S	Texture name

(V=Vector, F=Float, S=String)

TRAIN2

Suspension train on a straight track. The following parameters are supported:

Parameter	Type	Description
END1	V	First end point of track
END2	V	Second end point of track
HEIGHT	F	Height of suspension track over ground [m] Default: 11
MAXSPEED	F	Maximum speed of train [m/s] Default: 30
SLOWZONE	F	Distance over which train slows down at end of track [m] Default: 100
TEX	S	Texture name

(V=Vector, F=Float, S=String)

LPAD1

An octagonal bordered landing pad. Default diameter 80m (at scale 1). Landing pads are numbered in the order they appear in the list. Can be assigned numbers 1-9. For expected layout of texture map see e.g. Textures\Lpad01.dds.

Parameter	Type	Description
POS	V	Pad centre coordinates (in local coordinates of the surface base).
SCALE	F	Scaling factor. Default: 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX	S	Texture name. Default: none
NAV	F	frequency [kHz] of VTOL nav transmitter (valid range: 85.0-140.0, default: none)

(V=Vector, F=Float, S=String)

LPAD2

A square landing pad. Default size 80m (at scale 1). Landing pads are numbered in the order they appear in the list. Can be assigned numbers 1-99. For expected layout of texture map see e.g. Textures\Lpad02.dds.

Parameter	Type	Description
POS	V	Pad centre coordinates (in local coordinates of the surface base).
SCALE	F	Scaling factor. Default: 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX	S	Texture name. Default: none
NAV	F	frequency [kHz] of VTOL nav transmitter (valid range: 85.0-140.0, default: none)

(V=Vector, F=Float, S=String)

MESH

Generic mesh for custom object types. Mesh files must be in ORBITER mesh file format (see *3DModel.pdf* in the Orbiter SDK package).

Parameter	Type	Description
FILE	S	Mesh file name (without path and extension). Mesh files must be located in the mesh subdirectory (see master config file).
POS	V	Position of mesh origin (in local coordinates of the surface base).
SCALE	V	Scaling factors in x and z, and height in y. Default: 1 1 1
ROT	F	Rotation around vertical axis (degrees). Default: 0
TEX	S	Texture name. Default: none
UNDERSHADOWS		Object can be covered by shadows cast on the ground by other objects (e.g. roads, landing pads, etc.). Default: object not covered by ground shadows
OWNMATERIAL		Use materials and textures defined in the mesh file. This overrides the TEX entry.
LPAD		Object is a landing pad.
PRELOAD		Mesh should be loaded at program start. This can reduce disk activity during the simulation but increases main memory usage. Default: Load only when used.

(V=Vector, F=Float, S=String)

Notes:

- Generic meshes cannot cast shadows. This will be fixed in the future, but shadows will be required to have a convex hull.
- If the mesh only uses a single texture it is more efficient to specify it via the TEX entry than via the mesh using OWNMATERIAL, because Orbiter can merge objects with the same TEX entries for improved performance.

17.6 Scenario files

Scenarios contain all parameters required to set up the simulation at a particular time. They are used for loading and saving simulation states. Scenario files are usually generated automatically when saving a simulation. The format description below is primarily intended for developers of scenario editor add-ons.

Format:

```
<Description block>
<Environment block>
<Focus block>
<Camera block>
<HUD block>
<Left MFD block>
<Right MFD block>
<Station list>
<Ship list>
```

Description block (optional):

Contains a short description of the scenario.

```
BEGIN_DESC
  <Description>
END_DESC
```

<Description>: ASCII text describing the scenario. CR is converted to space. Empty lines are converted to CR.

Environment block (optional):

Contains the simulation environment.

```
BEGIN_ENVIRONMENT
  <Environment parameters>
END_ENVIRONMENT
```

<Environment parameters>:

Parameter	Type	Description
-----------	------	-------------

SYSTEM	S	Name of the planetary system. A configuration file for this system must exist. Default: "Sol"
DATE		Contains simulation start time. Allowed formats are: MJD <i><mjd></i> (<i><mjd></i> : Modified Julian Date) JD <i><jd></i> (<i><jd></i> : Julian Date) JE <i><je></i> (<i><je></i> : Julian Epoch) Default is current simulation time, but this should be avoided if the scenario contains objects defined by position/velocity vectors, which cannot easily be propagated in time.

Focus block (mandatory):

Contains parameters for the user-controlled spacecraft.

```
BEGIN_FOCUS
  <Focus parameters>
END_FOCUS
```

<Focus parameters>:

Parameter	Type	Description
SHIP	S	Name of the user-controlled ship. The ship must be listed in the ship list (see below).

Camera block (optional):

Camera mode and parameters. If the camera block is missing, the camera is set to cockpit view in the current focus object.

```
BEGIN_CAMERA
  <Camera parameters>
END_CAMERA
```

<Camera parameters>:

Parameter	Type	Description
MODE	Flag	Extern or Cockpit
TARGET	S	Camera view target. (external modes only; cockpit mode always refers to current focus object)
POS	V	Camera position relative to target (external modes only)
TRACKMODE	Flag [+String]	TargetRelative AbsoluteDirection GlobalFrame TargetTo <i><ref></i> TargetFrom <i><ref></i> Ground <i><ref></i> (external modes only)
GROUNDLOCATION	F F F	longitude (deg), latitude (deg) and altitude (m) of ground observer (<i>Ground trackmode only</i>)
GROUNDIRECTION	F F	polar coordinates of ground observer orientation (<i>free Ground trackmode only</i>)
FOV	F	Field of view (degrees)

HUD block (optional):

HUD mode and parameters. If the HUD block is missing, no HUD is displayed at startup.

```
BEGIN_HUD
  <HUD parameters>
END_HUD
```

<HUD parameters>:

Parameter	Type	Description
TYPE	Flag	Orbit Surface Docking

Left/Right MFD blocks (optional):

Left/right MFD type and parameters. If the block is missing, the corresponding MFD is not displayed. Note that custom MFD modes may have their own set of parameters.

```
BEGIN_MFD Left/Right
  <MFD parameters>
```

END_MFD

<MFD parameters>:

Parameter	Type	Description
TYPE	Flag	MFD type: Orbit Surface Map Launch Docking OAlign OSync Transfer
REF	S	Reference object (Orbit and Map MFD only)
TARGET	S	Target object (for Orbit, OAlign and OSync MFD only)
BTARGET	S	Base target (for Map MFD only)
OTARGET	S	Orbit target (for Map MFD only)
PROJ	Flag	Ecliptic Ship Target (for Orbit MFD only)
MODE	Flag	Intersect 1 Intersect 2 Sh periapsis Sh apoapsis Tg periapsis Tg apoapsis Manual axis (for OSync MFD only)
MANUALREF	F	Reference axis position [deg] (for OSync MFD in manual mode only)
LISTLEN	I	Number or orbit time listings (for OSync MFD only)



Station list:

With the support for vessel-vessel docking, stations should now be defined as vessels. Support for dedicated orbital stations is retained for backward compatibility, but may be dropped in a future version.

List of orbital stations. The list is optional, but all stations referred to by the ship list (see below) must exist.

```
BEGIN_STATIONS
  <Station 0>
  <Station 1>
  ...
  <Station n-1>
END_STATIONS
```

Station entries: <Station i>:

```
<Station name>[: <Class name>]
  <Station parameters>
END
```

<Station name>: station identifier string

<Class name>: station class (if applicable). If no class is specified, a .cfg file for the station, <station name>.cfg is required.

<Station parameters>:

Parameter	Type	Description
REF	S	Reference planet/moon which the station is orbiting
EL_MJD	F	Time in MJD format the orbital elements refer to
ELEMENTS	List	Orbital elements (ecliptic reference) consisting of 6 values: semi-major axis a [m], eccentricity e , inclination i [°], longitude of ascending node Ω [°], longitude of periapsis ϖ [°], mean longitude at EL_MJD L [°].
ZROT_FREQ	F	Rotation frequency for spinning stations. If not specified, the class value is used. Rotation is always around the station's local z-axis.
AROT	V	Orientation: rotation angles of object frame [°]

Ship list:

List of spacecraft. The list must at least contain the vessel referred to by the Focus entry.

```
BEGIN_SHIPS
```

```

<Ship 0>
<Ship 1>
...
<Ship n-1>
END_SHIPS

```

Ship entries <Ship i>:

```

<Vessel name>[: <Class name>]
  <Vessel parameters>
END

```

<Vessel name>: ship identifier string

<Class name>: vessel class (if applicable). If no class is specified, a .cfg file for the vessel, <vessel name>.cfg is required.

<Vessel parameters>:

Parameter	Type	Description
STATUS	Flag	Landed <planet> Orbiting <planet> Docked <planet>: <station>: <dock>
BASE		<base>: <lpad> (only for STATUS Landed)
HEADING	F	Orientation (only for STATUS Landed)
RPOS	V	Position rel. to reference (only for STATUS Orbiting)
RVEL	V	Velocity rel. to reference (only for STATUS Orbiting)
ELEMENTS	List	Orbital elements. This is an alternative to RPOS and RVEL for vessels with STATUS Orbiting. The list contains 7 entries: semi-major axis a [m], eccentricity e , inclination i [°], longitude of ascending node Ω [°], longitude of periapsis ϖ [°], mean longitude at reference date [°], and reference date in MJD format.
AROT	V	Orientation: rotation angles of object frame (only for STATUS Orbiting)
VROT	V	angular velocity [°/s] (only for STATUS Orbiting)
FUEL	F	Fuel level (0 to 1). This entry sets the level of all propellant resources to the same level. For individual settings, use PRPLEVEL option instead.
PRPLEVEL	List	List of propellant resource levels. Each entry is of the form <id>: <level>, where <id> is the resource identifier, and <level> is the propellant resource level (0..1).
THLEVEL	List	List of thruster settings. Each entry is of the form <id>: <level>, where <id> is the thruster identifier (in the order of thruster creation), and <level> is the thruster level (0..1). Thrusters with level 0 can be omitted.
DOCKINFO	List	Docking status list. This contains information about all docked vessels. Each entry is of the form <id>: <rid>, <rvessel> where <id> is the docking port identifier, <rid> is the docking port identifier of the docked vessel, and <rvessel> is the name of the docked vessel. Only occupied docking ports are listed. See notes below.

Docking vessels

There are two ways to define vessels as being assembled into a superstructure by docking them together:

- Place the vessels so that their docking ports coincide (by using appropriate RPOS, RVEL, AROT and VROT parameters for both). Orbiter will dock two vessels automatically if their docking ports are close enough.
- Define the DOCKINFO lists for both vessels so that they reference each other. Orbiter will then attach the vessels accordingly. **Important:** The RPOS, RVEL, AROT and VROT

parameters of the first vessel in the list which belongs to the superstructure are used to initialise the state vectors of the superstructure. All subsequent vessels docked to the same superstructure do not need to define these parameters.

17.7 Migrating from station to vessel definitions

Beginning with version 021201, vessels support multiple docking ports and can be docked to each other. There is no longer a need for dedicated space station classes. Instead, stations should be defined as vessels to make use of the new improved super-structure capabilities (for example, docked vessels behave correctly when one of the vessels fires its engines, while thrust forces are ignored for vessels docked to old-style stations).

Converting a station into a vessel only requires a few simple changes to its configuration and scenario entry.

In the configuration file (.cfg):

- Make sure that the `Mass`, `Size` and `Inertia` entries are present.
- You should usually set
`EnableFocus = FALSE`
to disable switching the input focus to the station (unless you explicitly want to make the station controllable)
- You should usually set
`EnableXPDR = TRUE`
`XPDR = <step>`
to equip the station with a homing beacon which assists approaching spacecraft in finding the station. `<step>` defines the frequency of the beacon and should be an integer between 0 and 639.
- In the dock-list, the approach directions (the second triplet of numbers in each row) must be inverted for technical reasons. For example,

```
BEGIN_DOCKLIST
13.67 5.56 0   1 0 0   0 1 0
END_DOCKLIST
```

becomes

```
BEGIN_DOCKLIST
13.67 5.56 0  -1 0 0   0 1 0
END_DOCKLIST
```
- In addition, an IDS (Instrument Docking System) can be defined for each dock by appending the frequency step to the end of the dock definition, for example:

```
BEGIN_DOCKLIST
13.67 5.56 0  -1 0 0   0 1 0   588
END_DOCKLIST
```

The IDS assists vessels during final docking approach. Again, the number must be between 0 and 639.

For more details on definition files for vessels, see the 3DModel document in the `Orbitersdk\doc` folder.

In the scenario file(s):

- Move the station entry from the `STATIONS` list to the `SHIPS` list.
- Replace
`REF <body>`
with
`STATUS Orbiting <body>`
to make the entry compatible with vessel scripts. `<body>` is the name of the planet or moon the station is orbiting.
- Combine the `EL_MJD` and `ELEMENTS` entries
`EL_MJD = <date>`
`ELEMENTS = <a> <e> <i> <Omega> <omegab> <L>`

into a single line:

```
ELEMENTS = <a> <e> <i> <Omega> <omegab> <L> <date>
```

- Go through the list of ships and check whether any of them are docked to the converted station via the `STATUS Docked` entry. If so, they must be converted to a vessel-vessel docking status using the `DOCKINFO` commands on both the vessel and the station (see Section *Scenario files* above).

Note that an object can be referenced by as a vessel in one scenario, and as a station in another scenario. It is therefore not strictly necessary to convert all scenario files.

18 Planet surface textures

18.1 Texture format

Each planet has an associated texture file *<pname>.tex*, where *<pname>* is the planet's name. The texture file contains a series of texture maps, stored as DirectDraw surfaces (dds) in DXT1 compression format.

ORBITER uses a variable resolution approach for both meshes and texture maps to render planetary surfaces. The rendering resolution level is determined by the apparent radius of the planet. At low resolutions ORBITER uses a single spherical mesh covered by a single texture. At higher resolutions the spherical surface is constructed from a series of sphere patches, each containing its own texture patch. This method allows efficient rendering by removing hidden patches before invoking the rendering pipeline.

ORBITER currently supports 8 resolution levels for planetary surfaces, as listed in Table 1. At the highest resolution the sphere is constructed from 364 patches with an effective texture resolution of 8192x4096. Figure 42 shows a detail of the Martian surface rendered at different resolution levels.

Level	Resolution*	Mesh patches	Triangles (total)**	Texture memory***	
				with DXT1	w/o DXT1
1	64 x 64	1	144	2K	16K
2	128 x 128	1	256	10K	80K
3	256 x 256	1	576	42K	336K
4	512 x 256	2	1024	106K	848K
5	1024 x 512	8	2592	362K	2.9M
6	2048 x 1024	24	4672	1.1M	9.0M
7	4096 x 2048	100	25440	4.3M	34.6M
8	8192 x 4096	364	116448	16.0M	127.8M

Table 1: Supported resolution levels for planetary surfaces.

*Resolution: Effective texture map resolution at the equator.

**Triangles: This is the total number of triangles for all patches. In practice fewer triangles will be rendered because hidden patches are removed before entering the rendering pipeline.

***Texture memory: Video/AGP memory required to hold texture maps up to this resolution level for a single planet. With DXT1: video hardware supports DXT1 texture compression.

W/o DXT1: video hardware doesn't support DXT1 texture compression.

High resolution levels require significant amounts of video/AGP memory and should only be used on systems with adequate 3D graphics subsystems. On older graphics cards which do not natively support DXT1 texture compression ORBITER needs to convert textures into RGBA format which increases memory requirements 8-fold. Conversion to RGBA will also dramatically increase the loading time when starting ORBITER.



Important: Do not try to use resolution level 8 if your video card does not support DXT1 texture compression or has less than 32MB of texture memory!

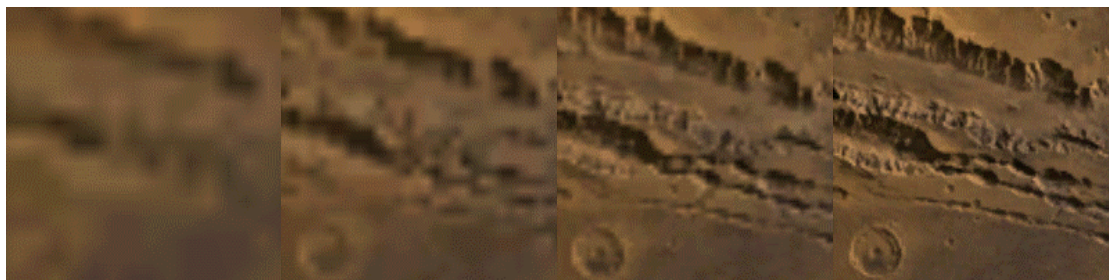


Figure 42: Mars texture detail at resolution levels 5, 6, 7 and 8 (from left).

18.2 Where ORBITER looks for textures

ORBITER first searches for the texture file in the location specified by the `HightexDir` entry in the *Orbiter.cfg* file (see section 17.1). If the texture file is not found or if `HightexDir` is not defined then ORBITER searches in the directory specified by the `TextureDir` entry. This allows switching between high and low resolution texture maps conveniently by inserting or removing the `HightexDir` entry.

If no texture file is found then the planet is rendered without a surface texture.

Each planet's configuration file *<pname>.cfg* contains an entry `MaxPatchResolution` which defines the maximum texture resolution level to use with this planet (valid range 1 to 8). If the texture file contains higher resolution levels than defined by `MaxPatchResolution` then the additional resolutions are skipped. This allows reducing texture memory requirements without modifying the texture file. If the texture file contains fewer resolution levels than defined by `MaxPatchResolution` then the maximum resolution is reduced accordingly.

18.3 Using *pltex* to generate custom planet textures

If you prefer, you can use your own planet maps instead of those provided by ORBITER. The ORBITER download page contains a planet texture conversion tool (*pltex*) which allows to convert planet maps from BMP bitmap format to ORBITER's texture format. It resamples the map to the requested resolutions, splits it into surface patches and converts them to DXT1 compressed texture format.

The source map should contain the complete surface in spherical projection, where the left edge corresponds to longitude 180° W, the right edge to longitude 180° E, the bottom edge to latitude 90° S, and the top edge to latitude 90° N. The width/height ratio of the bitmap should be close to 2/1.

Pltex requires the source map in 24bit or 8bit Windows BMP format. If your map is in any other format (e.g. JPEG or GIF) you need to convert it into BMP (using your favourite graphics conversion tool) before invoking pltex.

Synopsis:

```
pltex [-i <mapname>] [-l <minres> -h <maxres>]
```

<mapname>: source texture file name

<minres>: minimum resolution level (1..8)

<maxres>: maximum resolution level (*<minres>*..8)

- If command line options are omitted then pltex requests values interactively.
- If a higher maximum resolution is requested than can be obtained from the source map, pltex adjusts the maximum resolution accordingly. See Table 1 for map resolutions at the various resolution levels.
- The only justification for *<minres>* ≠ 1 is if you want to compose certain resolution levels from a different source map, e.g. generate Earth resolution levels 1 to 7 from a map that includes clouds, and level 8 from a map without clouds. In that case pltex must be run twice, and the output texture files concatenated.
- The option to use alpha (transparency) maps is intended for semi-transparent cloud maps.

Pltex will generate a texture file *<mapname>.tex*. If necessary, rename to *<pname>.tex* where *<pname>* is the planet's name, and copy to the `TextureDir` directory (usually "Textures") or `HightexDir` directory (usually "Textures2") (see section 17.1).

Note:

To generate resolution level 8 from a 8192x4096 planet map pltex requires at least 128MB of system memory.

Appendix A Solar System: Constants and parameters

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

A.1 Astrodynamic 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 2: 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	τ_A	499.004783806 (± 0.00000001) s
Gravitational constant	G	$6.67259 \text{ (}\pm 0.00030\text{)} \times 10^{-11} \text{ kg}^{-1} \text{ m}^3 \text{ s}^{-2}$
General precession in longitude		5028.83 (± 0.04) arcsec/Cy
Obliquity of ecliptic (J2000)	ϵ	84381.412 (± 0.005) arcsec
Mass: Sun / Mercury		6023600. ($\pm 250.$)
Mass: Sun / Venus		408523.71 (± 0.06)
Mass: Sun / (Earth+Moon)		328900.56 (± 0.02)
Mass: Sun / (Mars system)		3098708. ($\pm 9.$)
Mass: Sun / (Jupiter system)		1047.3486 (± 0.0008)
Mass: Sun / (Saturn system)		3497.898 (± 0.018)
Mass: Sun / (Uranus system)		22902.98 (± 0.03)
Mass: Sun / (Neptune system)		19412.24 (± 0.04)
Mass: Sun / (Pluto system)		$1.35 \text{ (}\pm 0.07\text{)} \times 10^8$
Mass: Moon / Earth		$0.012300034 \text{ (}\pm 3 \times 10^{-9}\text{)}$

Table 3: Primary constants

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

Table 4: 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.

A.2 Planetary mean orbits (J2000)

(Epoch = J2000 = 2000 January 1.5)

Planet (mean)	a [AU]	e	i [deg]	Ω [deg]	ϖ [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
Earth	1.00000011	0.01671022	0.00005	-11.26064	102.94719	100.46435

Mars	1.52366231	0.09341233	1.85061	49.57854	336.04084	355.45332
Jupiter	5.20336301	0.04839266	1.30530	100.55615	14.75385	34.40438
Saturn	9.53707032	0.05415060	2.48446	113.71504	92.43194	49.94432
Uranus	19.19126393	0.04716771	0.76986	74.22988	170.96424	313.23218
Neptune	30.06896348	0.00858587	1.76917	131.72169	44.97135	304.88003
Pluto	39.48168677	0.24880766	17.14175	110.30347	224.06676	238.92881

Table 5: Planetary mean orbits

A.3 Planetary orbital element centennial rates

(for the mean elements given above)

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

“ arcsecs
 Cy Julian century
 a Semi-major axis
 e eccentricity
 i inclination
 Ω longitude of the ascending node
 ϖ 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.

A.4 Planets: Selected physical parameters

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

Planet	V(1,0) [mag.]	Geometric albedo	Equatorial gravity [m/s ²]	Escape velocity [km/s]
Mercury	-0.42	0.106	3.701	4.435
Venus	-4.4	0.65	8.87	10.361
Earth	-3.86	0.367	9.780327	11.186
Mars	-1.52	0.15	3.69	5.027
Jupiter	-9.4	0.52	23.12 ± 0.01	59.5
Saturn	-8.88	0.47	8.96 ± 0.01	35.5

Uranus	-7.19	0.51	8.69 ± 0.01	21.3
Neptune	-6.87	0.41	11.00 ± 0.05	23.5
Pluto	-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.

A.5 Rotation elements

Planet	North pole		Obliquity of ecliptic* [°]	Longitude of Sun's transit* [°]
	Right ascension α_1 [°]	Declination δ_1 [°]		
Mercury	280.99	61.44	7.01	228.31
Venus	272.78	67.21	1.27	302.07
Earth	-	90	23.44	0
Mars	317.61	52.85	26.72	262.78
Jupiter	268.04	64.49	2.22	157.68
Saturn	40.14	83.50	28.05	349.39
Uranus	257.29	-15.09	82.19	167.62
Neptune	295.25	40.63	29.48	221.13
Pluto	311.50	4.14	68.69	225.19

Reference:

The Astronomical Almanac 1990 (North pole coordinates)

(*) Derived from north pole coordinates (MS)

A.6 Atmospheric parameters

Planet	Surface pressure [kPa]	Surface density [kg/m ³]	Scale height [km]	Avg. temp [K]	Wind speeds [m/s]
Mercury					
Venus	9200	~65	15.9	737	0.3-1 (surface)
Earth	101.4	1.217	8.5	288	0-100
Mars	0.61 (variable)	~0.020	11.1	~210	0-30
Jupiter	>> 10 ⁴	~0.16 at 1 bar	27	~129 ~165 at 1 bar	up to 150 at < 30° latitude up to 40 else
Saturn	>> 10 ⁴	~0.19 at 1 bar	59.5	~97 ~134 at 1 bar	up to 400 at < 30° latitude up to 150 else
Uranus	>> 10 ⁴	~0.42 at 1 bar	27.7	~58 ~76 at 1 bar	0-200
Neptune	>> 10 ⁴	~0.45 at 1 bar	19.1-20.3	~58 ~72 at 1 bar	0-200
Pluto					

Appendix B 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
Ω	Longitude of ascending node
ω	argument of periapsis
ν	true anomaly

B.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:

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

Eccentricity:

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

Inclination:

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

Longitude of ascending node:

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

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

Ω 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:

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

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

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

ω 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.

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

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

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

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:

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