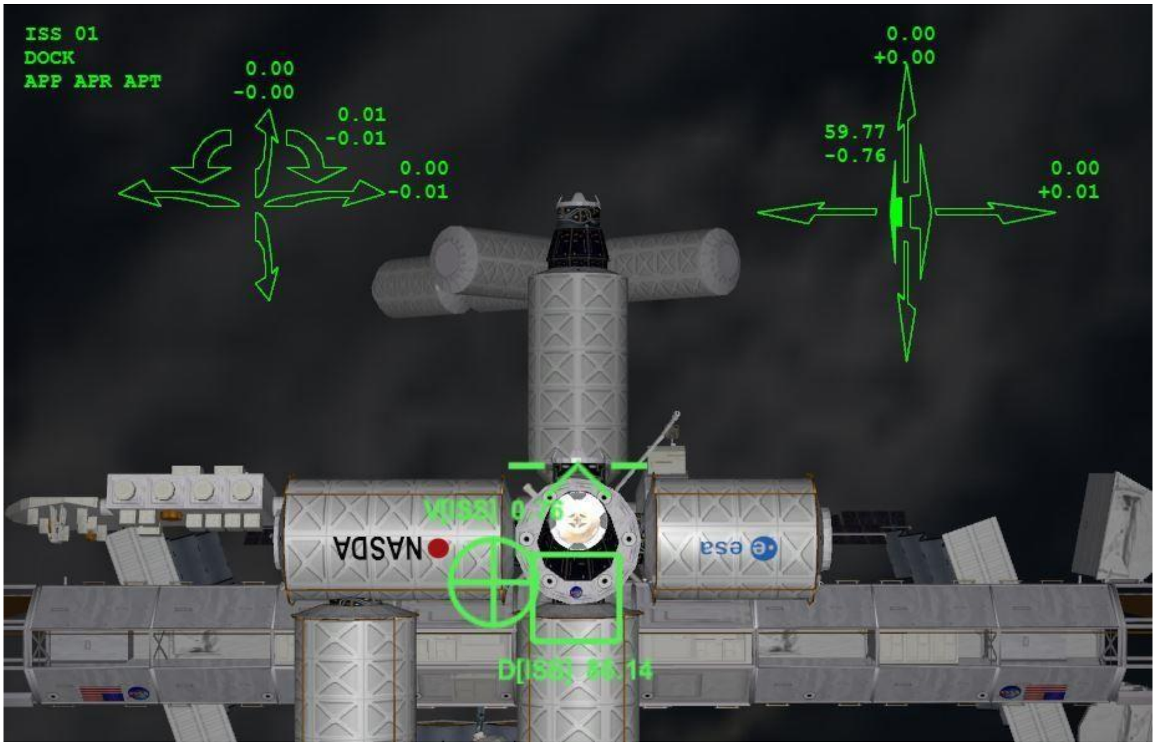
Rendezvous Orientation 3.05 MFD/HUD User Documentation



Introduction

RV Orientation MFD & HUD is a utility to assist with alignment and docking to target vessels. Key features include:

- A Head-Up Display that integrates with the standard Docking HUD making it much richer
- A fuzzy-logic autopilot capable of flying in partial or fully automatic mode to dock
- A series of guidance modes to align you to specific points on the approach corridor
- A spherical waypoint system to guide you to a safe docking when you are behind your target port
- Detailed offsets and rates to allow you to fly absolutely precise manual or semi-manual approaches
- A self-calibration system to tune the autopilot to handle the flight characteristics of any vessel

Credits

I'd like to credit Szymon "Enjo" Ender for his Module Messaging, HUD Drawer and MFD Button Page code which are extensively used in this MFD, and to Martin Schweiger for the Orbiter environment and simulator. Thanks also to those responsible for hosting and moderating the Orbiter Forum and Orbit Hangar sites that bring the community together, and to my fellow add-on developers for your numerous extensions to this amazing simulation environment.

Mandatory Pre-Requisites

1. HUDDrawer SDK v.0.4 or later

This is Enjo's HUD hooking code. Please download from orbithangar.com at http://www.orbithangar.com/searchid.php?ID=6023. If you have errors about a missing VesselHooking.dll when starting Orbiter with RV Orientation active, then that's because this prerequisite is not installed.

GPL Copyright

What does this mean in simple terms? This utility is free for you to use and distribute, with credits please, and it has all the source code included for you to modify as you wish. (See http://www.gnu.org/philosophy/free-sw.html for the philosophy of "free software"). But if you do modify it, you are required to credit this original code and to release your work also as free software, and (most critically) you must include the whole source code so others can recompile and do further awesome things to it in the future. Free and fair for all, letting the code live if you move away from it sometime later.

Basic Concepts

To fly a safe and accurate approach, you first need to get to the vicinity of your target (say within 20km). You can do this with standard MFD's – e.g. Align Planes and Sync Orbit – or with more advanced MFD's such as TransX or IMFD. I'll assume you know how to do this, but if not, read the free "Go Play In Space" document for a great primer.

Once in the vicinity of your target, you will need to point to the negative relative velocity (RVEL) vector (the circle with the dot in it), and thrust your vessel forward until the RVEL drops close to zero. This puts you “on station”, holding position to the target. You are now able to focus on moving into close alignment ready for docking.

To execute the docking maneuver, you need to move to a point some distance directly in front of the target docking port, and then orient your ship to point directly to the target, with your rotation aligned to the target port’s local “up” direction. If you are used to using the Docking HUD, then this alignment is achieved when you are aligned with the docking approach corridor, with the bar at the top of each rectangle. (In this RV Orientation HUD, you can use Docking HUD in combination with the Docking HUD, or replacing it, as you prefer.)

Once aligned at this ‘primary waypoint’ at the end of the docking approach corridor, you need to fly the approach towards the target docking port, all the time maintaining your pitch, yaw and roll (or bank), up/down, and left/right alignment. Easy right? Actually no, because both you and the target are usually moving in 3D space, and small movements on the target make the end of a docking corridor move about quite a bit (unless you switch on an autopilot on the target, or null out all the attitude rates with Scenario Editor). And sometimes – e.g. for the Luna space station, the port will be rotating, so you need to match yourself to the rotation as well.

Now – we also need to consider how to fly an appropriate course to get to this ‘primary waypoint’, avoiding flying straight through the target vessel, and also avoiding collisions with extended objects like solar arrays. In other words, you should maintain a safe spherical distance to the target vessel at all times until you are on the final docking approach corridor. This also applies when you start the docking maneuver inside this safe distance, or from behind the target port.

This MFD / HUD will plot a safe course to do this maneuver, and optionally can fly it semi-automated or fully-automated.

Introducing the RV Orientation modes

RV Orientation has three modes of operation: RVEL, SWP, and DOCK.

RVEL (Relative Velocity): the Relative Velocity is the vector and speed that you need to go to hold a neutral distance from your target in 3D space. This mode guides you towards the negative RVel direction, so you can apply forward thrust or main engine thrust to zero out the RVel. It's useful for accurate alignment with the RVel as you come on station from your rendezvous burn, with optional autopilot support if you prefer. Note: it's just a rotation mode to orient you to the negative RVel direction, so there is no concept of translation in this mode.

SWP (Spherical Waypoint): this mode plots a course through a number of intermediate waypoints to your primary waypoint on the final approach to your target port, whilst maintaining a minimum separation around the target vessel. You can visualize this as a safety sphere around the target vessel. The rotation guidance continually points you directly at the target port, with port alignment up, and the translation guidance moves you to each waypoint in turn. As you near the waypoint, the navigation system will auto-select the next waypoint in sequence, until you arrive at the primary waypoint directly in front of your target port.

Waypoints are expressed in degrees of virtual latitude and longitude from the target port, taking the port's up-down axis to define the virtual Prime Meridian (longitude 0), and the left-right port axis defining the virtual Equator (latitude 0). The waypoints are aligned at 30 degree points of longitude, and 15 degree points of latitude, with the exception of the final longitude point before the primary waypoint, which is at +/- 15 degrees. E.g. a sample course starting high and back-left of the target port could take you through the following SWP lat-long waypoints: SWP.60.-150, SWP.60.-120, SWP.60.-90, SWP.45.-60, SWP.30.-30, SWP.15.-15, SWP.0.0, and from there into the final waypoint alignment with the docking corridor.

In order to optimize the final turn from going around the safety sphere to going into the docking port, the range around the sphere is approximately 250m (800 ft) larger than the approach waypoint distance. This makes a nice turn in to the primary waypoint.

DOCK (Docking): this mode controls the final approach through to docking. It guides you along the navigation line from the primary waypoint to the dock, maintaining a good orientation and alignment.

Introducing the Autopilot

The autopilot on RV Orientation has been designed to operate with as many vessel classes as possible, in as many configurations and weights as possible. It implements a fuzzy logic control loop aimed at driving a negative relative velocity rate according to the size of the error. (In other words, if your positional error is positive, then the velocity should be negative to reduce the error, and it should be proportional to the magnitude of the positional error). As with all automation, it is meant to be an aid to the pilot flying, and not a replacement. I.e. please step in and override the auto pilot at any time you are not happy with its performance. The autopilot generally performs well up to 10x time compression, but you should consider dropping to real time (1x time compression) for sharp turns to minimize the overshoot.

WARNING: You MUST have your thrusters in a mode that aligns with your ship's docking port. E.g. for the XR-5, this mode is "RCS Docking Config". The autopilot assumes you are looking in the direction of your docking port, so “up”, “forward” etc are all relative to the port’s orientation, NOT the vessel’s orientation. If you enable the autopilot on a ship with a docking port not aligned with front of your vessel, and the autopilot is doing crazy things, then this is why.

There are three autopilot modes:

APR (Autopilot Rotate): the autopilot controls Pitch, Yaw, and Roll (Bank) orientation, leaving you free to focus on the translation alignment and range. (Note, you can run APR and APT together).

APT (Autopilot Translate): the auto pilot controls Up-Down and Left-Right alignment, leaving you to work on orientation and range. This mode is also useful if your target is in free drift (i.e. not holding an orientation) and is gently turning. In this case, the extended waypoints movement is exacerbated by range, and you want to let the autopilot focus 100% on getting to the waypoint rather than on maintaining orientation.

APP (Approach Autopilot): this enables the full autopilot. It requires APR and APT to also be active (and auto-activates them if not). This autopilot selects SWP mode, flies the whole waypoint path to the primary waypoint, then aligns to the docking corridor, then runs in DOCK mode through to contact. The approach autopilot has numerous safety features on the approach; constantly checking it is positioned within an alignment cone that gets smaller as distance to dock reduces. Likewise, it constrains the orientation into an ever-decreasing error tolerance. If the ship slides out of the alignment cone or orientation tolerance, it will hold range whilst correcting the error. If the ship is considered badly out of alignment or orientation, it backs off to a safe range to retry. For example – if you try to fly the approach at 100x time compression, most likely you will not be able to dock, as the errors from running at such high speed will take the vessel outside of the safe alignment when in close proximity to the target vessel, so it will back you off to a safe range. (Hint … don’t try to dock at 100x time compression!)

There are two further autopilot utilities, discussed here:

APC (Autopilot Calibration): The stock RV Orientation is pre-calibrated for the Delta Glider, DG-S, DG-IV, DG-XR1, XR-2, XR-5, ShuttleA, PLV and CEV-Orion. If you wish to use it with a new vessel, use this APC mode to calibrate the autopilot for it. Note – you can also force-override the calibration of an existing calibrated vessel if you want (e.g. because you are flying with cargo or attachments that significantly changes the angular momentum on the rotation thrusters. To use this APC mode, you need to be in the vicinity of a target vessel, with your ship in the desired configuration for dock maneuvering (eg in your dock view, with thrusters aligned with your dock). The ship will warp you out to around 100 meters from your target, then execute a series of predefined calibration burns to determine the acceleration values in each dimension. Each of these burns is done a number of times, on heavy fuel and light fuel, to create a calibration matrix for the full control algorithms. The calibration takes around 10 minutes, and is a one-off thing for this vessel type. Once the calibration is complete, you will be returned to your previous position and fuel state. The results of a calibration run are stored in .\CONFIG\MFD\RVO\RVO_Rates.cfg. If you do calibrate a new vessel, please consider sending me the extra line added to that rates file, and I'll merge into the next distribution (with accreditation).

APD (Auto Pilot Dump): This is a diagnostic utility. This writes an RVO_AP_Dump.csv file for analysis of autopilot behavior. Hit APD to activate, then hit again to stop, and then you can open the CSV file.

The fields are for technical analysis and debug. For anyone interested in each column, then this is the decode:

SimT - simulation time
StepAvg - average step duration of the past 10 intervals (for calculating thrust durations)
Dx, Dy, Dz ... directions in degrees (rotation about X = Pitch, about Y = Yaw, about Z = roll)
Px,Py,Pz ... positions in meters (X = right, Y = up, Z = fwd)

Dx = Pitch offset in degrees
Dx.V = Velocity (rate) of the pitch offset in degrees/sec
Dx.L = AP fuzzy logic determined limit for the rate of pitch in degrees/sec
Dx.Th = Thruster magnitude (-1.00 to 1.00) for pitch up (-ve for down)
Dx.C = Estimated sim cycles to attain desired offset at full thrust
Dx.A = Reference calibrated acceleration of pitch in degrees/sec^2 for our current weight at full thrust
Dx.B = AP fuzzy logic accumulator bias (used to nudge the rate when very low thrust needed)

Same applies to Dy, Dz, Px, Py, Pz.

Pz.Tgt overrides the Z direction target to hold the ship or back it off when out of cone.

AppInCone = 1 when approach is in the allowed cone
MaxDirErr = degrees of orientation error permitted at this range
MaxPosErr = meters of translation error permitted at this range

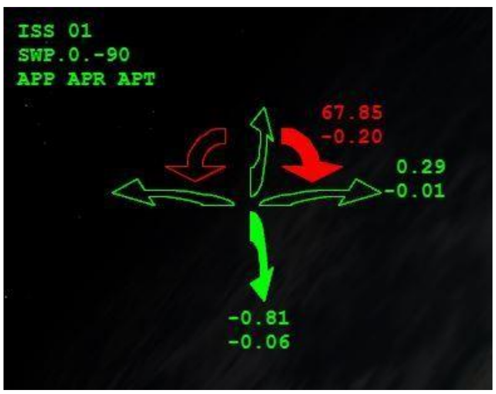
RotActive = AP working on ROT in this cycle
AttActive = AP working on ATT this cycle (interlocked to not do both on same cycle)
AppActive = AP working on FWD/BACK this cycle
AppAttAllow = Interlock to prevent ATT mode when ROT errors are too big
ApToggle = Counter for how many ATT cycles per ROT when we are settled down

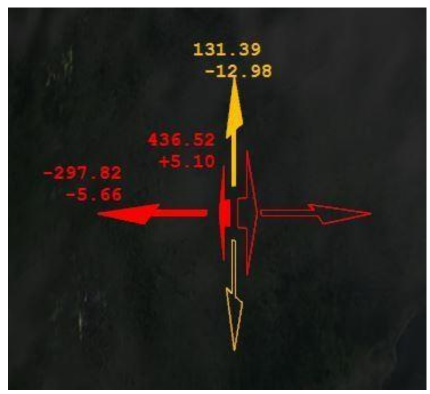
Range = 3D spherical range from target
Our Lat, Long = Lat Long from target port's frame to our port
WP Lat, Long = current waypoint in SWP mode

Mode = -1 for AP Calibrate, 0 for dock, 1 for WP, 2 for SWP, 3 for RVEL.

The original reason for this mode was to be able to dump the autopilot control responses in real time, whilst running the simulation. If you ever try debugging the code in the middle of a thruster sequence, you quickly find out that the 2 minutes you have been staring at some code usually translates to your ship now completely out of control, miles off from its target!

The HUD





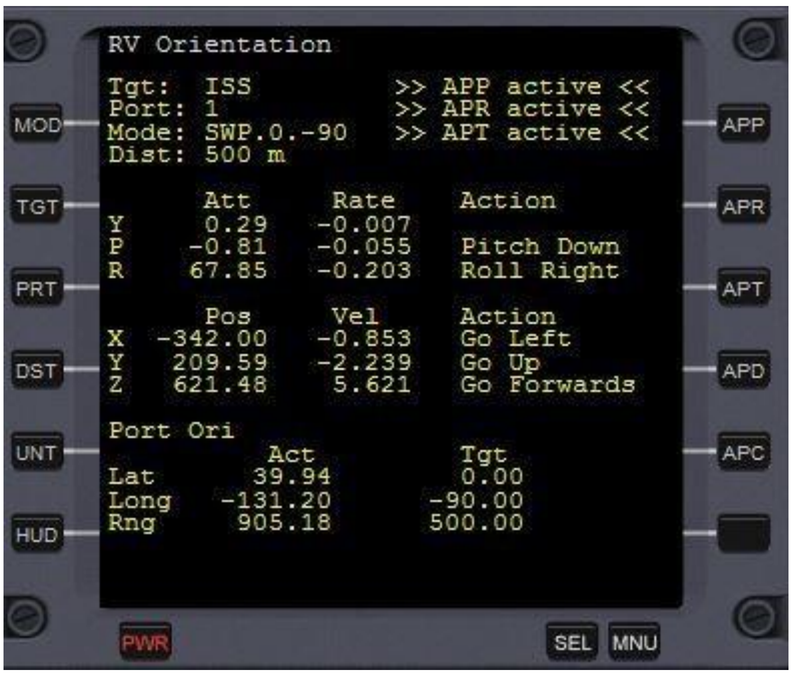
One of the differentiating features on this MFD is the Heads-Up Display (or HUD). There is no formal interface to the HUD in Orbiter, but thanks to a great piece of code by ENJO (HUD Drawer SDK, which is a required dependency), there is a way to access the HUD for the majority of vessels.

RV Orientation takes advantage of this HUD Drawer interface to paint orientation and translation data on the display in a visual and numerical format. For the visual format - the display has color-coded arrows indicating direction to move and amount of error. Green is good, amber is for you to recognize the misalignment, and red is when you are quite a long way off alignment. On final approach, the forward/back color coding indicates rate – i.e. green means that you are coming in at an appropriate speed, amber is a warning, and red indicates you are making people really nervous on your target ship! For the numerical format, the arrows are annotated with error amount and rate for more accuracy.

Note: if the HUD does not come up on your vessel, then check it on a default vessel such as a Delta Glider. If it does not work there either, you have not enabled the HUD Drawer correctly. If it does work on a Delta Glider, but not on your vessel, then the vessel needs coding modernization changes that can be done, but are beyond the scope of this document. (For example: CEV Orion does not support the HUD hooks, so I cannot display anything on that HUD).

For small size HUDs – e.g. the 3D Virtual Cockpit mode of the Delta Glider – this version of RV Orientation repositions the elements to make maximum use of the HUD real estate. For example – the docking mode goes to the bottom right, the arrows are moved down and inwards, and the rates are repositioned accordingly.

The Other MFD Buttons



MOD (Mode): Selects the mode as above (i.e. RVEL, SWP, DOCK)

TGT (Target): Prompts you to enter the target vessel, and then auto-selects the first free port for docking. Note – if LaunchMFD has a target, and DV orientation doesn't, then it will auto-import the target in from LaunchMFD. If you have RV orientation up first, then bring up LaunchMFD and select target, then RV Orientation will auto-import it on the next MFD launch (e.g. press F8, or bring up RV Orientation again in an External MFD).

PRT (Port): Selects the next free port on the target vessel. It loops through all available ports then back to the first again.

DST (Approach Distance): Sets the length of the approach corridor from the primary waypoint to the target port, and the radius of the safety sphere for the SWP mode.

UNT (Units): Sets the units to US (i.e. feet and ft/sec) or METRIC (i.e. meters and m/s). This applies to the HUD and MFD displays, the scenario quick save information, and to the DST entry.

General Usage Tips

As with any MFD or utility, the goal is to give you the pilot more enjoyment and control of your vessel. This utility allows you to do some amazing docking maneuvers in full control. For example - if you want to diagonally translate onto the target from one meter away, then this MFD will let you do this with 1 cm of accuracy. Or if you want to do a 'grand tour' of docking to every port on the ISS in record time, then set the approach distance to 100m and use the APP to move from port to port. (Tip: change the port when 1m out and you can see the APP autopilot deal with the new request!)

There are some edge cases where you may need to intervene to help the auto pilot. If your target is moving around, and your relative velocity is greater than say 100m/s, and your range is say 20km or more, you may want to take more direct action with main engine to null things out and move closer, as the auto pilot works on thrusters only. Likewise, for aesthetics, the auto pilot wants to keep you pointing at your target. This works well when you are under control, but the auto pilot may need to focus 100% on translation to get closer rather than aesthetics, particularly for wobbling targets.

Remember, as a responsible and alert pilot, you should always be ready to override the automation any time you are not happy with the situation.

Some Development Notes

This MFD & HUD has approximately 4,500 lines of code. It took around 6 months from original concept to this release, and around 500 hours of actual work to create. Without Enjo’s HUD Drawer code, this would all not have been possible, and I also make extensive use of his button handling code too.

I've created a little framework for persisting data per vessel, per MFD position, per combination of vessel and MFD position and globally for RV Orientation, so you can shut down and restart RV Orientation, change MFD position, have different settings in parallel on multiple vessels, resize external MFD's and all those good things without losing context. It helps me think about the context that I am working – e.g. would another MFD want to see this information, and what should happen when I switch to a different vessel.

The auto pilot was 60% of the challenge. I originally implemented a PD algorithm (see PID control loops e.g. on Wiki), assuming because it worked on my usual XR-5 at a specific weight, then it would work the same at any XR-5 weight and on any other vessel. I started to realize the enormity of the difference between realistic ships and say the Delta Glider, and how critical it would be to calibrate the response very carefully. This led to the APC mode and the idea of running multiple passes on the same thruster banks to get an average response. I initially plotted out lots of calibration runs for different vessels and weights, and I accidentally discovered that there is an almost perfect power-series correlation between thruster acceleration and ship weight, of the form: $y = Ax^B$. By differentiating this to $\ln(y) = \ln(A) + B.\ln(X)$, this power relationship becomes a straight-line graph, of slope B and intersect ln(A). So I run the calibration with fuel tanks full and empty (leaving a bit in the thruster tanks!), and this gives me two data points on this line to make a really accurate interpolation and extrapolation of thruster performance for all weights. The only concern is when changes to the mass also change the angular momentum (e.g. if you extend a satellite to full length on an arm), but I'm assuming that you would not be doing this whilst trying to dock on autopilot!

If you do delve into the code, and anything doesn't make sense, then please PM me and I'll see if I can explain it to you.

Andrew "ADSWNJ" Stokes – July 2013, updated November 2013, May 2016, and August 2018