



# RXP GNS 530W/430W V2

## User's Manual For Prepar3d / Flight Sim

### Fly Waas Approaches!

The WAAS-certified GNS 530W and its slightly smaller sibling, GNS 430W, lead the industry with multitasking, integrated avionics and cutting-edge WAAS navigation. The GNS 530W comes with optional Class B TAWS alerting to warn you of potential terrain and obstacle conflicts along your flight path. Both come with optional Ryan 9900B TCAD, Shadin Fuel and Shadin Air input, and cross fill capability.

The Reality XP GNS 530W/430W V2 is an entirely new GNS WAAS simulation product built on our unique expertise in designing the most authentic Garmin devices simulation since 2002.

### About This Manual

This manual guides you through the product features and is intended for flight simulation purposes only. It shall not be used for any real world aviation application or reference. By reading this manual, you should become well acquainted with the product, and should be able to obtain the information necessary to use the product in your simulator.

We thank you for having chosen a Reality XP Product and wish you a pleasant and a safe virtual flight with us.

**Product disclaimer: this software is designed for entertainment only.** Although we have designed the product to resemble and function like the original, it is not designed as a training device. Not all systems have been simulated, and some of those that have been simulated may not be entirely functional.

**NOT FOR USE IN REAL FLIGHT OR AIRCRAFT OPERATION.**

<b>INTRODUCTION</b>	<b>3</b>
Garmin Reference Guides	3
System Requirements	3
<b>GETTING STARTED</b>	<b>4</b>
Installation	4
Add-Ons Menu	4
Configuration Panel	5
Using The Mouse	5
Resizing Gauge Windows	5
<b>ADD-ON DETAILS</b>	<b>6</b>
Autopilot	6
Radios	6
Traffic Collision Alerting Device (Tcad)	6
Taws-B	6
Integral Lighting	6
Cross Fill	6
Satellite-Based Augmentation Systems (Sbas)	7
Discrete Inputs	7
Flight Simulator Gps Commands	7
Keyboard Shortcuts	7
Additional Gauges	7
Failures Annunciation	8
Hardware Controllers	8
Aircraft Flight Manual Supplement (Afms)	8
Checklists	8
Flight Plans And User Waypoints	9
<b>CONFIGURATION</b>	<b>10</b>
Aircraft Configuration	10
Advanced Settings	10
Gauge Rendering Settings	11
Third Party Vendor Settings	11
Keyboard Shortcuts	12
Custom Sim Variables	13
Custom Gps Commands	14
Device Embedded Memory	16
Configuring Alternate Paths	16
Configuring Aviation Database Files	17
Installing Multiple Garmin Trainers	17
<b>TROUBLESHOOTING</b>	<b>18</b>

## Introduction

The Reality XP GNS 530W and GNS 430W V2 gauges is a faithful reproduction that pilots and flight simulator users can use it as a training tool to familiarize themselves with the workings of the actual equipment. Each button and knob is fully functional and performs identically to its real-world counterpart.

Garmin avionics have become the clear #1 choice in the owner-flown aircraft market. Nearly three-fourths of all U.S. single- and twin-engine piston and turbine aircraft retrofitted since 2000 are now flying with Garmin GPS-enabled equipment. The provision of the integrated communication and navigation radios in the unit gives the extra capabilities of auto tuning and pre-fetching your navigation and approaches frequencies. Flying has never been easier!

The Reality XP GNS 530W and GNS 430W V2 are navigator's heaven with an extraordinary number of user customizable navigation information fields. In short, there is simply no better simulation of the GNS 530W and GNS 430W for any flight simulator, period.



### Garmin Reference Guides

In addition to this user's manual which guide you through the functions within the simulator, we recommend referring to the original Garmin GNS series pilots and cockpit reference guide that is installed in its own Garmin program group. Please take the time to read all manuals completely so that you can become properly acquainted with the product and its operation.

### System Requirements

The Reality XP GNS 530W/430W V2 is an add-on for **Prepar3d v4/3/2/1**, **Flight Simulator X - Steam Edition**, **Flight Simulator X** or **Flight Simulator 2004** for Windows<sup>1</sup>.

---

<sup>1</sup> Versions required: Prepar3d 4 (64bits), P3D3, P3D2, P3D1, FSX-SE, FSX-ACC/SP2, FS9-SP1; Windows versions required: WinXP to Win10

## Getting Started

### Installation

An add-on with both<sup>2</sup> the GNS 530W and the GNS 430W simulation installs to<sup>3</sup>:

My Documents\Prepar3d v4 Add-ons\Reality XP GNS\

During installation you will be prompted to download and install the Garmin GNS Trainer Lite<sup>4</sup>. The first time you install this package, or should you need to repair the Garmin Trainer installation, the installer will save a copy of the trainer installer in the following folder:

C:\ProgramData\Reality XP\Common\GnsTrainer\

This Garmin trainer is a more recent version never offered before in flight simulation, exclusively showcased in our solution. It has main software version 3.3 released in 2010 and includes bug fixes, faster updates, takes less resources and adds some new capabilities to the integrated navigator.

### Add-Ons Menu

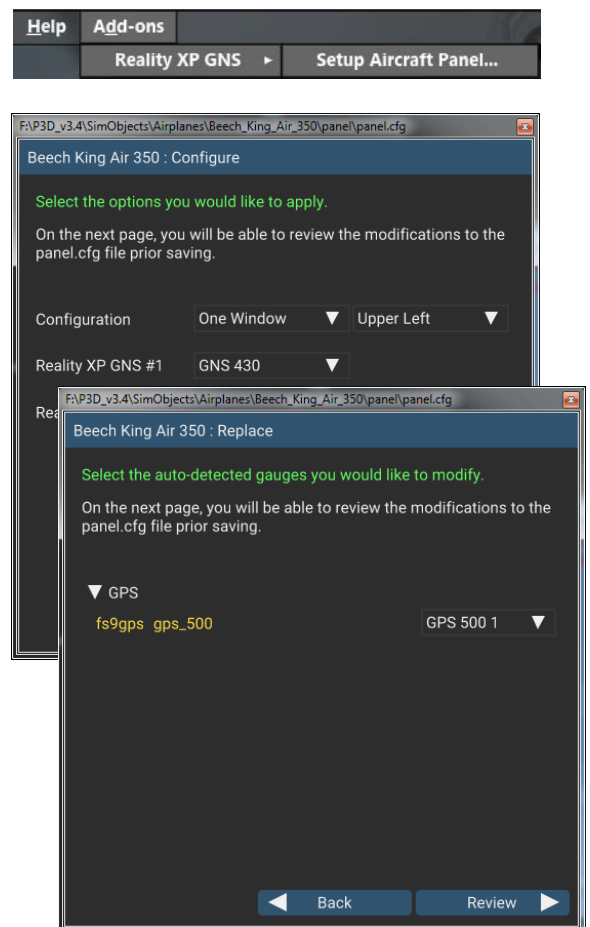
The GNS 530W/430W V2 Menu loads automatically and is operated using the Prepar3d/FSim add-ons menu<sup>5</sup>.

Select the **menu Add-ons | Reality XP GNS** to add or remove GNS units to your aircraft panel. Once selected, the panel wizard opens and guides you through. This task-oriented context aware assistant can help you with the following tasks:

- Configure your aircraft panel with GNS popup windows.
- Replace any similar GPS gauge with a GNS gauge using an auto-detect feature.
- Remove all GNS gauges from the panel.
- Restore the panel to its prior state using the automatic backup.

The first time the assistant modifies the panel, it saves a backup copy of the panel.cfg file next to the original in the same folder. The last option restores the panel from this backup.

Should you manually modify the panel.cfg file afterward, make sure to make your own copy prior using the 'Restore' function otherwise the manual modifications will be lost.



<sup>2</sup> The same gauge file installs both but it requires purchasing and installing both products to unlock both GNS models.

<sup>3</sup> P3D3: "Prepar3d v3 Add-ons"; P3D2, P3D1, FSX-SE, FSX, FS9: in their respective "Gauges" folder.

<sup>4</sup> Compatible trainer versions: 3.3

<sup>5</sup> FS9 only: the menu displays in a popup when right-clicking anywhere on the FS9 window. See Troubleshooting chapter for other simulators.

### Configuration Panel

The GNS can be tailored to suit your needs by configuring the way each device looks and behaves. Most of each device configuration options are available from the configuration panel.

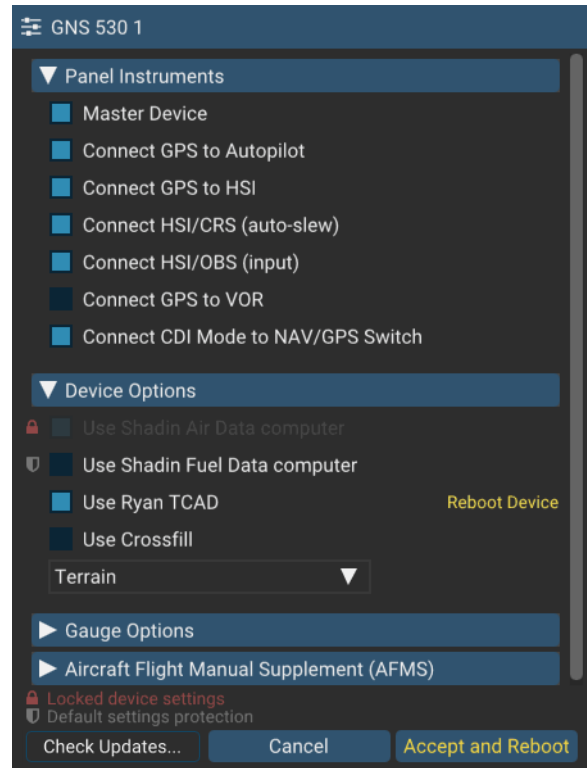
**Press and hold the SHIFT key then RIGHT-Click the top edge** of the GNS device to open the configuration panel.

Options are set up for an individual aircraft and then saved to the "RealityXP.GNS.ini" file in the aircraft folder. The add-on restores the last known settings when loading the aircraft.

Settings with **a lock or a shield icon** have restrictions. Read more about this in the *Third Party Vendor Settings* chapter.

The 'Reboot Device' tag appears when a setting requires rebooting the device.

Select **Check Updates...** to verify whether you are running the latest version.



**⚠ Drag and Drop** any RealityXP.GNS.ini file onto the panel window to import another aircraft settings.

### Using The Mouse

This gauge implements a relatively unique implementation of click spots. They work as follows:

- The mouse cursor contextually changes depending on the possible actions.
- As the mouse cursor passes over a click spot it changes from an arrow cursor to a “hand” cursor.
- Whenever a single click spot is used, a left click will accomplish the task.
- Left clicks turn knobs to the counter-clockwise direction when the mouse is in an area to the left of the knob and to the clockwise direction when the mouse is in an area to the right of the knob.
- Mouse wheel turns knobs in either direction regardless if the mouse is to the left or to the right.

### Resizing Gauge Windows

The simulation window resizes and maintains its aspect ratio. **Press and hold the SHIFT key then LEFT-Click the top edge** of the GNS device to scale the window to either 1:1 or 2:1 screen sizes.

## Add-on Details

The add-on includes a complete collection of Garmin GNS devices: the **GNS 530** product includes the **GPS 500**, similar in form and function to the 530, it is a radio-less sibling and fits panels already equipped with radios. Similarly, the **GNS 430** product includes both the **GPS 400** and the **GNC 420**.



### Autopilot

The GNS is capable of precise course deviation and roll steering outputs that can be coupled to the simulator autopilot so that IFR flight procedures may be flown automatically. It controls the simulator A/P system both horizontally and vertically.

### Radios

The GNS 530W and GNS 430W V2 can control the simulator COM and NAV audio panel volumes using the **Power** and the **Volume** knobs respectively (P3D2.5+ only).

### Traffic Collision Alerting Device (TCAD)

The GNS embeds a Ryan 9900B TCAD (Traffic Collision Alerting Device) simulation. It is an authentic TCAS II Mark 7 simulation passing the 250 TSIM tests. Take full control of the TCAD detection shields, view nearby traffic and take actions upon visual and aural TA advisories directly on the GNS device. The Flight Simulator AI Traffic simulation variables feed the TCAD simulation.

### TAWS-B

The GNS V2 offers both TAWS-B and TERR Proximity modes with advanced settings available to fine tune the capabilities depending on the aircraft configuration. The GNS also connects to the flaps and the gear in order to adjust detection sensitivity and operating limits depending on the phase of flight.

### Integral Lighting

Using the configuration panel, select whether you prefer constant or adjustable brightness, independently for the screen and the bezel. The later uses custom simulation variables third party aircraft vendor can use.

### Cross Fill

When enabled in both GNS, this option allows you to transfer a direct- to destination, the active flight plan including VNAV parameters, any stored flight plan or user waypoints to a second 400W or 500W device.

If both units are set to automatic (in the Garmin GNS device settings), a change in the active flight plan, or VNAV parameters, on one unit can also be seen in the other. Initiating a direct-to course to a waypoint on one unit also initiates a direct-to course to the same waypoint on the other unit.

### Satellite-Based Augmentation Systems (SBAS)

The GNS supports **WAAS** (US), **EGNOS** (EU), **MSAS** (JP) and **GAGAN** (IN)<sup>6</sup>. Display the last GNS AUX page, select **SBAS Selection** then enable each system and improve the performance of the GNSS for enroute, terminal area, non-precision approach operations (including "GPS" or "RNAV" approaches), and approach procedures with vertical guidance (including "**LNAV/VNAV**" and "**LPV**").



### Discrete Inputs

Each device supports discrete inputs, via keyboard shortcuts, custom simvars or the default GPS commands. These control some of the GNS operation: Toggle CDI source, Toggle OBS mode and TAWS inhibit.

### Flight Simulator GPS Commands

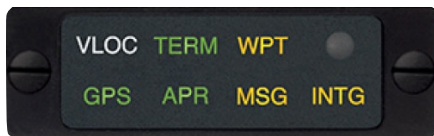
The GNS may be configured to intercept all of the default FS GPS commands. Open the settings panel then select "Use Default GPS Commands" advanced gauge option.

### Keyboard Shortcuts

Each device supports mapping their buttons and knobs to keyboard shortcuts. These commands are described in more details in the 'Configuration' chapter.

### Additional Gauges

In addition to the main GNS gauges, the product installs additional **Mid-Continent gauges** ready to be added into any panel<sup>7</sup>. These are fully working replicas displaying valuable information on the panel and including a convenient and functional **TERR INHB** button.



MD41-1484W Annunciator Control Unit



MD41-1028 TAWS Control Unit


<sup>6</sup> Systems and approaches availability depends on the navigation database.

<sup>7</sup> Ready-to-use with the pre-installed add-on.xml on P3D4 and P3D3.

See "C:\Program Files (x86)\Reality XP\Gauges\GPS Annunciation\rxpGpsAnnun (readme).rtf" for more information.

## Failures Annunciation

When enabled in the configuration panel, the GNS plugin uses the following sim variables<sup>8</sup> as failures sources:

rxp.failures.gps_1 rxp.failures.gps_2	Satellite reception failure, displays: "No GPS Position"	
rxp.failures.waas_1 rxp.failures.waas_2	Differential GPS failure, reduces location accuracy: satellite reception mode switches to 3D only.	
rxp.failures.raim_1 rxp.failures.raim_2	RAIM check failure.	
PARTIAL PANEL AVIONICS	Shuts down unit's power.	

NB: professional licenses provide additional datarefs to override GPS receiver mode, WAAS solution, dilution of precision, position uncertainty and figure of merit.

## Hardware Controllers

The GNS supports a selection of external hardware controllers<sup>9</sup>. When the add-on loads, it automatically detects the connected hardware and saves the bindings to the "RealityXP.GNS.ini" file in the aircraft folder. The following hardware devices are supported: PFC 430 Control Head (Standalone and included in PFC Avionics Stack).

## Aircraft Flight Manual Supplement (AFMS)

The GNS supports specific enablement features configured during installation in a real aircraft. These are regrouped in the configuration panel under Aircraft Flight Manual Supplement (AFMS) and allow selecting V-Flag Declutter, disabling CDI key etc...

## Checklists

The Checklists function provides a built-in method of reviewing your aircraft checklist. As each Checklist is completed, you can advance to the next one in order. Checklists are created using any text editor and must be saved to a file named **checklistst.gns**<sup>10</sup> into the aircraft folder. To get started, a sample pack installs to:

```
"C:\ProgramData\Reality XP\Common\GnsTrainer\checklists.zip"
```

Unzip the "checklists.zip" archive file to extract the sample ".txt" checklists file. Open the unzipped checklist.txt with notepad.exe to review the file format (it is self-documented). Then copy the file to the aircraft folder and rename it to checklist.gns.

On the GNS device, whether on the last boot page or while viewing the AUX pages group, select the **Checklists** menu to start the checklists function.

<sup>8</sup> Variables starting with 'rxp' can be set from either XML or C++ gauges like any other 'L' var. See 'Control Variables' p13 for complete list.

<sup>9</sup> Requires separate activation or a hardware enabled version. See reality-xp.com for available options. Existing customers of the GNS Driver don't have to purchase the driver a second time.

<sup>10</sup> Supports both "checklist.txt" and "checklist.gns"



## Flight Plans And User Waypoints

Flight plans can be created on a computer using compatible flight planning software and saved to a virtual datacard folder to be imported into the GNS. The imported flight plans can then be stored to the GNS flight plan catalog. This chapter describes how to enable and use this feature.

### Preparing the virtual datacard

The first step is to create a virtual datacard folder where to store the flight plans and the user waypoints. This folder must be named **FPL** and be located in the GNS Trainer Data folder, typically<sup>11</sup>:

```
"C:\ProgramData\Garmin\GNS Trainer Data\GNS\FPL"
```

The second step is to create one or more flight plan files with the ".fpl" file extension. The product comes with a starter pack installed into the following folder:

```
"C:\ProgramData\Reality XP\Common\GnsTrainer\flightplans.zip"
```

Alternatively, this could be a "FPL" shortcut instead of a folder, linked to any other folder on the hard drive, for example:

1. Create a folder "Documents\GNS Flight Plans" where to copy the flight plan files and user waypoints file.
2. Open "C:\ProgramData\Garmin\GNS Trainer Data\GNS\", right-click the empty area to the right, select **new | shortcut**
3. Name the shortcut "FPL" and select the browse button
4. Browse to select the "Documents\GNS Flight Plans" folder and press OK.

### Importing flight plans

The last step is to import the flight plans into the GNS device. To proceed with the starter pack, unzip the "flightplans.zip" archive file to extract the "LFATLFRG.fpl" flight plan sample file, then copy the file to the "FPL" folder.

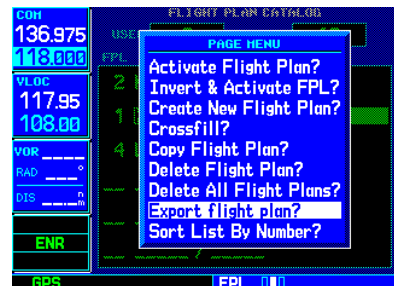
On the GNS device, while viewing the Flight Plan pages, turn the outer GPS knob clockwise to reach the **3rd FPL page** and select flight plan file, and the FPL number this flight plan will be assigned to. When done, select **Import?**



### Exporting flight plans

The GNS offers a feature to export the flight plans from the device. On the GNS device, while viewing the Flight Plan Catalog page, press the GPS knob to activate the cursor and select a flight plan in the list.

Then press the **Menu** key and select the **Export Flight Plan?** option. The exported flight plan is saved to the same "FPL" folder with a unique file name.



<sup>11</sup> The "FPL" folder must reside in the folder equivalent to "GNSAPPDATA" (see Configuring Alternate Paths chapter for more information).

## Configuration

### Aircraft Configuration

This chapter details the general organization of the settings file. The GNS settings are automatically saved in the file **RealityXP.GNS.ini** located in the aircraft folder. Every time a GNS opens from the Reality XP GNS menu, it creates this self-documented settings file on save.

The INI file consists of sections and keys. The name of a section is always entered inside the square brackets. Each section contains several keys (the key must be always assigned to the section that begins in the file before this key). The content of one section ends by a definition of the next section. The name of a key is always on a separate line and the content is assigned by the = char. A line comment starts with the ; char.

```
[GNS_530_1]
; is the master device if true.
MasterDevice = true
; selects COM/NAV Radios '1' or '2'.
RadiosPair = 1
; connects GPS to HSI if true.
LinkHsi = true
; connects GPS to HSI course if true.
LinkCrs = true

[GNS_530_1.DEFAULT]
; show screen only gauge if true.
nobezel = false
; enable mouse clickspots if true, disable if false.
usemouse = true
; left mouse button popup ident (ex: GPS_PANEL or 225).
popleft =
```

There are multiple sections in the Reality.GNS.ini file:

- The **GNS\_530\_1** gauge section contains the settings for this device.
- The **GNS\_530\_1.DEFAULT** section contains the default settings for each device instance.

### Advanced Settings

Advanced settings are only manually editable in the **RealityXP.GNS.ini** settings file in a **[GNS\_###\_#]** section:

CpuAffinity = -1	selects the CPU running the trainer process: 1 to 32; 0: last core, -1: automatic.
HardwareDevice =	; connects to: 'PFC_STACK', 'PFC_430', 'PFC_530' or any '#PID' (#D002 for PFC_430) no value connects to first found, 'OFF' disables connection (requires the Reality XP GNS Driver add-on)
HardwareIdx = -1	; hardware selector number or -1 to disable (requires the Reality XP GNS Driver add-on)

### Gauge Rendering Settings

A Flight Simulator gauge is a template which defines what to display, and the panel.cfg tells where gauges display. Each entry of the same gauge is an instance (a sibling) from the same template (parent). For example, a GNS 530 both in a popup window and the VC Panel is a single gauge with two instances. When adding multiple instances of the same gauge to the panel.cfg file, the **RealityXP.GNS.ini** supports per-instance configuration.

The automatically generated RealityXP.GNS.ini file in the aircraft folder has **[GNS\_###\_#.DEFAULT]** sections for each device found in the panel. Each gauge sibling uses the default settings from this section, for example [GNS\_530\_1.DEFAULT]. Should you want a particular instance of the gauge (one sibling) to use other than 'DEFAULT' settings, a new named section must be created and the **gauge 5th parameter**<sup>12</sup> must be set to this new section name.

For example, to set the VC instance to render without a bezel (screen only): copy and paste the gauge 'DEFAULT' section to the end of the RealityXP.GNS.ini file, change the suffix from 'DEFAULT' to 'NO\_BEZEL' then edit the new section settings.

The **RealityXP.GNS.ini** file may be configured like this:

```
[GNS_530_1.NO_BEZEL]
; show screen only gauge if true.
nobezel = true
...
```

The **panel.cfg** file may be configured like this:

```
[VCockpit01]
gauge00=rxpGNS2!GNS_530_1, 0, 0, 100, 90, NO_BEZEL
...
```

In the example above, the 5th param was arbitrarily named 'NO\_BEZEL'<sup>13</sup>. However, you can create as many uniquely named sections as needed, and choose a relevant suffix to match the 5th param:

```
[GNS_###_#.MY_NICE_VC]
[GNS_###_#.MY_COOL_POPUP]
```

### Third Party Vendor Settings

Third party aircraft vendor may want to **lock settings** and prevent user changes in the GUI settings panel, or to ensure the device starts up with **default values** like pre-configured 3D cockpit rendering position and size. When configured, locked settings appear disabled with a red lock icon in the GUI settings panel. Besides, default settings appear with a gray shield icon and reset to their default value every time the device reloads.

**Default Settings:** create a **RealityXP.GNS.defaults.ini** file in the aircraft folder with only the sections, keys and values you want to override as defaults.

**Locked Settings:** manually add the following to the **RealityXP.GNS.ini** settings file in a **[GNS\_###\_#]** section. For example, an aircraft requires the device to be set to 'com1/nav1' and connected to the HSI:

```
locked = RadiosPair, LinkHsi      comma separated list of read only settings. (NB: case sensitive names)
```

---

<sup>12</sup> gauge\_file ! gauge\_name, x, y, w, h, **param**

<sup>13</sup> Make sure to use non space separated words.

## Keyboard Shortcuts

This chapter describes the list of custom commands available to control the GNS from keyboard shortcuts. Each command exists with the suffix "\_1" for unit #1 or "\_2" for unit #2. For example, the command name for the device #2 "HOME" button is: "HOME\_2".

To change hotkeys, open the "My Documents\Reality XP\GNS Simulation\FitSim\RealityXP.GNS.ini" file with notepad. It is a self-documented configuration file that lets you configure any combination of ALT, CTRL, SHIFT, WIN and key per device button and knob.

CFLP	press button: COM fLIP	PWR_PUSH	press knob: POWER
VFLP	press button: NAV flip	PWR_CCW	rotate knob: POWER (ccw)
CDI	press button: CDI	PWR_CW	rotate knob: POWER (clock-wise)
OBS	press button: OBS	VOL_PUSH	press knob: VOLUME
MSG	press button: MSG	VOL_CCW	rotate knob: VOLUME (ccw)
FPL	press button: FPL	VOL_CW	rotate knob: VOLUME (clock-wise)
VNAV	press button: VNAV	COM_PUSH	press knob: COM
PROC	press button: PROC	COM_OUTR_CCW	rotate outer: COM (ccw)
ENT	press button: ENT	COM_OUTR_CW	rotate outer: COM (clock-wise)
CLR	press button: CLR	COM_INNR_CCW	rotate inner: COM (ccw)
MNU	press button: MNU	COM_INNR_CW	rotate inner: COM (clock-wise)
DTO	press button: DTO	GPS_PUSH	press knob: GPS
RDN	press button: RING UP	GPS_OUTR_CCW	rotate outer: GPS (ccw)
RUP	press button: RING DWN	GPS_OUTR_CW	rotate outer: GPS (clock-wise)
CDI_SOURCE	toggle CDI source (discrete input)	GPS_INNR_CCW	rotate inner: GPS (ccw)
OBS_MODE	toggle OBS mode (discrete input)	GPS_INNR_CW	rotate inner: GPS (clock-wise)
TAWS_INHIBIT	inhibit TAWS (discrete input)	TOGGLE_WINDOW	toggle window visibility

In addition the add-on exports the following additional commands designed to work across multiple Reality XP plugins:

RXP_GPS_MASTER_DEVICE_0	select master device: default GPS
RXP_GPS_MASTER_DEVICE_1	select master device: Reality XP GPS 1
RXP_GPS_MASTER_DEVICE_2	select master device: Reality XP GPS 2
RXP_GPS_MASTER_DEVICE_NEXT	select next available master device

## Custom Sim Variables

This chapter describes the list of additional sim variables the GNS publishes to third party add-ons. There are three sets of variables representing different Reality XP sources: any GPS #1 (GTN or GNS V2), any GPS #2 and any Reality XP Master Device. Variable names consists in a base name and a suffix indicating the source:

Variable Name (any RXP GPS)	Units	Suffix	Remarks
rxp.hsi.bearing_deg_mag	degrees	_1 _2 _pilot	GPS#1, GPS#2, Master Device
rxp.hsi.flag_from_to	number	_1 _2 _pilot	
rxp.hsi.hdef_dots	number	_1 _2 _pilot	
rxp.hsi.vdef_dots	number	_1 _2 _pilot	
rxp.hsi.has_dme	number	_1 _2 _pilot	
rxp.hsi.dme_distance_nm	miles	_1 _2 _pilot	
rxp.hsi.dme_speed_kts	knots	_1 _2 _pilot	
rxp.hsi.dme_time_sec	seconds	_1 _2 _pilot	
rxp.hsi.flag_glideslope	number	_1 _2 _pilot	
rxp.hsi.display_horizontal	number	_1 _2 _pilot	
rxp.hsi.display_vertical	number	_1 _2 _pilot	
rxp.gps.course_degtm	degrees	_1 _2	no suffix for Master Device rxp.gps vars
rxp.gps.cross_track_nm	miles	_1 _2	
rxp.gps.time_sec_eta	seconds	_1 _2	
rxp.gps.discrete_out	number	_1 _2	
rxp.gps.nav_id	number	_1 _2	6 chars encoded nav ident <sup>14</sup>
rxp.gps.unit	enum	_1 _2	4 chars encoded device id or 0 <sup>15</sup>
<b>Variable Name (RXP GNS V2 only)</b>			
rxp.radios.indicators.gns_unit1	enum		4 chars encoded device id or 0
rxp.radios.indicators.gns_unit2	enum		4 chars encoded device id or 0

## Control Variables

Third party gauges write to the following sim variables to further enhance the GNS V2 operation:

rxp.panel.rheostat_screen	number	(0 to 100)
rxp.panel.rheostat_integral	number	(0 to 100)
rxp.failures.gps_1(_2)	failure enum	(FAIL_NONE=0, FAIL_INOP=1, FAIL_BLANK=2)
rxp.failures.waas_1(_2)	failure enum	(FAIL_NONE=0, FAIL_INOP=1, FAIL_BLANK=2)
rxp.failures.raim_1(_2)	failure enum	(FAIL_NONE=0, FAIL_INOP=1, FAIL_BLANK=2)
rxp.gps.dscrt_in.obs_mode_sel_1(_2)	boolean	discrete in signal (write 1 to trigger)
rxp.gps.dscrt_in.taws_inhib_1(_2)	boolean	discrete in signal (write 1 to trigger)
rxp.gps.dscrt_in.cdi_src_1(_2)	boolean	discrete in signal (write 1 to trigger)
rxp.gps.dscrt_in.mute_actv_caut_1(_2)	boolean	discrete in signal (INOP)

<sup>14</sup> Because the SDK doesn't allow setting 'string' variables, the ICAO is encoded as a sequence of 6 chars into a 'double' (as binary data).  
C code example: `double id = get_named_variable_value(...); const char* szId = (const char*) &id;`

<sup>15</sup> **GTN:** 'G621' (0x31323647), 'G622' (0x32323647), 'G631' (0x31333647), 'G632' (0x32333647), 'G651' (0x31353647), 'G652' (0x32353647), 'G721' (0x31323747), 'G722' (0x32323747), 'G751' (0x31353747), 'G752' (0x32353747)  
**GNS:** 'G401' (0x31303447), 'G402' (0x32303447), 'G421' (0x31323447), 'G422' (0x32323447), 'G431' (0x31333447), 'G432' (0x32333447), 'G501' (0x31303547), 'G502' (0x32303547), 'G531' (0x31333547), 'G532' (0x32333547)

### 3D Models Variables

Each device shares the following variables to help synchronizing older 3D models with GNS V2 gauges<sup>16</sup>:

```
rxp.GNS_1.knob_powr          number (rotation angle in degrees)
rxp.GNS_1.knob_volm
rxp.GNS_1.knob_left
rxp.GNS_1.knob_rite
```

### Custom Gps Commands

The GNS intercepts standard Flight Simulator GPS commands with: **Settings Panel | Use Simulator GPS Commands.**

These commands are a set of named events which any other gauge, module or 3D model can trigger.

GPS_NEAREST_BUTTON	GNS CDI.
GPS_OBS_BUTTON	GNS OBS.
GPS_MSG_BUTTON	GNS MSG.
GPS_MSG_BUTTON_DOWN	GNS MSG (press).
GPS_MSG_BUTTON_UP	GNS MSG (release).
GPS_FLIGHTPLAN_BUTTON	GNS FPL.
GPS_VNAV_BUTTON	GNS VNAV.
GPS_PROCEDURE_BUTTON	GNS PROC.
GPS_ZOOMIN_BUTTON	GNS zoom in.
GPS_ZOOMOUT_BUTTON	GNS zoom out.
GPS_DIRECTTO_BUTTON	GNS Direct.
GPS_MENU_BUTTON	GNS Menu.
GPS_CLEAR_BUTTON	GNS CLR.
GPS_CLEAR_BUTTON_DOWN	GNS CLR (press).
GPS_CLEAR_BUTTON_UP	GNS CLR (release).
GPS_ENTER_BUTTON	GNS ENT.
GPS_POWER_BUTTON	GNS POWER/VOL Knob.
GPS_ACTIVATE_BUTTON	GNS NAV VOL Knob.
GPS_CURSOR_BUTTON	GNS push cursor.
GPS_GROUP_KNOB_INC	GNS GPS/CRS chapter up (right outer knob cw).
GPS_GROUP_KNOB_DEC	GNS GPS/CRS chapter dn (right outer knob ccw).
GPS_PAGE_KNOB_INC	GNS GPS/CRS page up (right inner knob cw).
GPS_PAGE_KNOB_DEC	GNS GPS/CRS page dn (right inner knob ccw).
GPS_BUTTON1	GNS COM/NAV toggle.
GPS_BUTTON2	GNS COM/NAV coarse up (left outer knob cw).
GPS_BUTTON3	GNS COM/NAV coarse dn (left outer knob ccw).
GPS_BUTTON4	GNS COM/NAV fine up (left inner knob cw).
GPS_BUTTON5	GNS COM/NAV fine dn (left inner knob ccw).
GPS_TERRAIN_BUTTON	GNS TAWS Inhibit (discrete in).

The GNS also supports the following standard Flight Simulator commands:

COM_STBY_RADIO_SWAP	GNS COM1 FLIP/FLOP.
NAV1_RADIO_SWAP	GNS NAV1 FLIP/FLOP.
COM2_RADIO_SWAP	GNS COM2 FLIP/FLOP.
NAV2_RADIO_SWAP	GNS NAV2 FLIP/FLOP.

---

<sup>16</sup> Replace rxp.GNS\_1.xxx with rxp.GNS\_2.xxx for the 2nd unit variables.

GNS V2 can share older Reality XP GNS Legacy vars with: Settings Panel | Use Legacy 3rd Party Aircraft Panel (L:vars)

### Command Modifiers

Flight Simulator commands have a parameter which is generally only used for actions requiring a value, like the position of a flap lever. Other actions which toggle or trigger a state change usually pass the value 0. The GNS uses this existing parameter in order to modify the "KEY\_GPS" (C++) or the "GPS" (XML) command action. This enhances integration between the GNS and 3D cockpit models, add-ons modules, any other XML or C gauges. The command parameter is interpreted as a bit-field where each binary digit represents a specific modifier:

- Bit #2: press and hold the button if set.
- Bit #3: release the button if set.
- Bit #4: command to GPS unit #2 if set, to GPS unit #1 if clear.

Sending the **MSG** key **press-release** event to **GPS#1: GPS\_MSG\_BUTTON, 0**

Sending the **MSG** key **press-release** event to **GPS#2: GPS\_MSG\_BUTTON, 8** = *bit4 (8)*

Sending the **ENT** key **press and hold** event to **GPS#2: GPS\_ENT\_BUTTON, 10** = *bit2 (2) + bit4 (8)*

Sending the **ENT** key **release** event to **GPS#2: GPS\_ENT\_BUTTON, 12** = *bit3 (4) + bit4 (8)*

**Exceptions:** **GPS\_POWER\_BUTTON** and **GPS\_ACTIVATE\_BUTTON** commands support "press-release" events only. To command knob rotation, use 'press and hold' for 'TURN LEFT' (CCW) and 'release' for 'TURN RIGHT' (CW).

### ModelDef.xml Example

A complete canonical implementation installs to:

"C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\modeldef (RealityXP GNS).xml".

Here is how to implement a typical **<PartInfo>** for the GNS V2 Device #2 "Clear" button.

```
<Name>RXP_GNSV2_CLEAR_BUTTON_2</Name>
<MouseRect>
  <Cursor>Hand</Cursor>
  <TooltipText>Clear / Cancel (Press and Hold)</TooltipText>
  <MouseFlags>LeftSingle+LeftRelease+DownRepeat</MouseFlags>
  <CallbackCode>
    (M:Event) 'LeftSingle' scmp 0 == if{ 10 (>K:GPS_CLEAR_BUTTON) }
    (M:Event) 'LeftRelease' scmp 0 == if{ 12 (>K:GPS_CLEAR_BUTTON) }
  </CallbackCode>
</MouseRect>
```

### Device Embedded Memory

The GNS Trainer stores all user settings in a file similarly to a real device using embedded non-volatile memory. Settings include user preferences such as displayed navigation units, map settings, routes catalog and the user waypoints. They are saved in the following folder:

```
C:\ProgramData\Garmin\GNS Trainer Data\GNS
```

The Reality XP GNS simulation manages individual non-volatile memory file copies, one per-device and per-unit. These are saved in the same folder next to the original file, using the following file name convention:

```
sys_nand0          : default original file.  
sys_nanda, sys_nandb: GNS 5XX #1 and GNS 5XX #2  
sys_nandc, sys_nandd: GNS 4XX #1 and GNS 4XX #2
```

Should you want to backup and restore the settings, you just need to make a copy of the corresponding file(s) listed above to a safe location, then copy them back to the aforementioned folder to restore.

The simulation automatically creates and manages per-device/per-unit nonvol file. In addition, should the original `sys_nand0` file is missing, the GNS Trainer recreates it from scratch. When upgrading from one trainer version to the next, the setting files are upgraded automatically as well.

### Configuring Alternate Paths

The Garmin trainer installs various components, each one in a specific path. This chapter describes supported Windows environment variables<sup>17</sup> to configure alternate trainer paths in a typical scenario.

Set the general trainer application path, that is where the trainer is installed:

```
GNSSIMPATH="C:\Program Files (x86)\Garmin\GNS Trainer\GNS"
```

Set the application data path, that is where it stores the non volatile memory files (where routes, user waypoints and device settings are saved)<sup>18</sup>:

```
GNSAPPDATA="C:\ProgramData\Garmin\GNS Trainer Data\GNS"
```

Set the application database path, that is where to look for all the db files (terrain, obstacle, navigation etc.):

```
GNSNAVDATA="C:\Program Files (x86)\Garmin\GNS Trainer\GNS\DB"
```

---

<sup>17</sup> how to add environment variables:

<http://www.kunal-chowdhury.com/2016/09/windows-10-anniversary-update-how-to-edit-environment-variables.html>

<sup>18</sup> The GNSAPPDATA default to the path shown If not manually configured otherwise.



### Configuring Aviation Database Files

Copy the aviation database file and/or the obstacle database file to the location of your choice on your hard drive, for example in "My Documents" folder, then add the following Windows environment variable for the aviation and obstacles files respectively (in this example the aviation db file is renamed to gns\_db.bin):

```
GNSADBFIL="C:\Users\MyName\Documents\gns_db.bin"           (alias: GNSDBFILE)
GNSODBFIL="C:\Users\MyName\Documents\terrain.odb"        (alias: GNSOBSTCL)
```

### Installing Multiple Garmin Trainers

This chapter describes a procedure when you want or need to install two different versions of the Garmin GNS Trainer on your computer. The Garmin GNS Trainer installer prevents installing two different versions, such as the latest compatible with the Reality XP GNS and an older one. However, the GNS plugin can be configured to use alternate trainer path and data path. Please note the following for dual Trainer installation:

1. Install the trainer for the Reality XP GNS
2. Copy the "GNS" folder located in "C:\Program Files (x86)\Garmin\GNS Trainer\" to the location of your choice, or rename it for example to: "C:\Program Files (x86)\Garmin\GNS Trainer\GNS (3300)\"
3. Add the following Windows environment variables:

name = "GNSSIMPATH" value = *the path to the 'GNS' folder you have chosen above*

for example:

```
GNSSIMPATH="C:\Program Files (x86)\Garmin\GNS Trainer\GNS (3300)\"
```

4. You can now uninstall the trainer and reinstall the old one: the GNS plugin will use the trainer and its data from the location indicated in your environment variables.

## Troubleshooting

This chapter describes solutions available to help troubleshooting problems with the add-on and the Garmin Trainer.

### Missing Add-On Menu

The Gauge Add-Ons menu requires **SimConnect** which is normally installed automatically with the simulator. For 32 bits simulators, it requires a minimum of SimConnect FSX-SP2. The installer file, **SimConnect.msi** is located on the original installation disks (FSX-ACC/SP2) or in the following folders:

#### Prepar3D:

P3D\redist\interface\FSX-SP2-XPACK\retail\lib\SimConnect.msi

#### FSX-SE:

SteamLibrary\steamapps\common\FXS\SDK\Core Utilities Kit\SimConnect SDK\LegacyInterfaces\FSX-XPACK

Should the menu fails to display, you can create or manually edit the add-on/SimConnect files with Windows notepad.exe

**P3D4:** open notepad, create a new text document, copy/paste the following content, then save the file as<sup>19</sup>:

My Documents\**Prepar3D v4 Add-ons**\Reality XP GNS\add-on.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<SimBase.Document Type="AddOnXml" version="3,3" id="add-on">
  <AddOn.Name>Reality XP GNS</AddOn.Name>
  <AddOn.Description>Legendary Gauges Redefined</AddOn.Description>
  <AddOn.Component>
    <Category>DLL</Category>
    <Name>Reality XP GNS Menu</Name>
    <Path>C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\64\rxpGNS2_menu.dll</Path>
  </AddOn.Component>
  <AddOn.Component>
    <Category>Gauges</Category>
    <Name>Reality XP GNS Gauge</Name>
    <Path>C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\64</Path>
  </AddOn.Component>
</SimBase.Document>
```

**P3D3:** open notepad as above but change both <Path> lines '64' to '32' and save the file as:

My Documents\**Prepar3D v3 Add-ons**\Reality XP GNS\add-on.xml

```
<Path>C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\32\rxpGNS2_menu.dll</Path>
<Path>C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\32</Path>
```

<sup>19</sup> make sure the file extension is .xml and not .txt

**P3D2, P3D1, FSX-SE, FSX:** the menu uses the global DLL.xml file found here:

C:\Users\[name]\AppData\Roaming\Microsoft\FSX\dll.xml

Open this file with notepad, copy/paste the following **<Launch.Addon>** **</Launch.Addon>** section<sup>20</sup>:

```
<?xml version="1.0" encoding="UTF-8"?>
<SimBase.Document Type="Launch" version="1,0">
  <Descr>Launch</Descr>
  <Filename>dll.xml</Filename>
  <Disabled>False</Disabled>
  <Launch.ManualLoad>False</Launch.ManualLoad>
  <Launch.Addon>
    ...
  </Launch.Addon>
  <Launch.Addon>
    <Name>Reality XP GNS Menu</Name>
    <Disabled>False</Disabled>
    <ManualLoad>False</ManualLoad>
    <Path>C:\Program Files (x86)\Reality XP\GNS Simulation\FltSim\32\rxpGNS2_menu.dll</Path>
  </Launch.Addon>
</SimBase.Document>
```

### Log Files

The solution monitors certain key operations and logs runtime data to specific files. When troubleshooting a problem, the first step is to analyze runtime data saved to the log files.

The gauge log file: "**My Documents\rxpGNS2.gau.log**"

The GNS Simulation log file: "**My Document\rxpGnsSim.dll.log**"

The add-on menu log file: "**My Documents\rxpGNS2\_menu.dll.log**"

### Debug Settings

The GNS simulation support additional settings to help troubleshooting video card drivers, and to report internal Garmin trainer errors. When enabled, the trainer will log additional information to "rxpGnsSim.dll.log"

The following keys must be manually added in the **RealityXP.GNS.ini** settings file in a **[GNS\_###\_#]** section:

LogTrainer = 0	1 to enable Garmin Trainer logs.
----------------	----------------------------------

---

<sup>20</sup> Everything included within **<Launch.Addon>** and **</Launch.Addon>** is GNS specific, the rest is the standard DLL.xml file content.

### **Anti Virus / Windows Security / Windows Defender Security Center**

Windows security and Antivirus or Malware scanners are known to flag the Reality XP files as dangerous from time to time. Should the product doesn't work as expected, or refuses to run, please make sure to review your anti-virus protection for any 'RXP' quarantined files. The solution is to recover the file(s) from the quarantine of the antivirus and to set an exclusion for it so it will not be deleted anymore.

### **How to allow blocked apps by the Windows 10's ransomware protection**

Windows Security lets you manage all your security needs, including Windows Defender Antivirus and Windows Defender Firewall. Controlled folder access can help prevent ransomware and other destructive malware from changing your personal files. Sometimes apps that you normally use might be blocked from making changes to common folders like Documents.

Starting with the Windows 10 October 2018 Update (version 1809), the feature has updated with a new option to quickly allow recently blocked applications without the need to disable the security feature just because of a false positive.

To access Controlled folder access controls, select Start > Settings > Update & Security > Windows Security > Virus & threat protection. You should then select Manage settings, and under Controlled folder access, then select Manage Controlled folder access. From there set Controlled folder access to on or off.

Within Windows 10 Windows Security you can now allow a recently blocked app by opening "Virus & Threat protection" within the Windows Security app, clicking "Ransomware protection" and then going through to "Allow an app through Controlled folder access". The new option to add "Recently blocked apps" will appear in a popup.

*This manual is intended for Flight Simulation use only, and may not be used in any real world aviation applications. The authors are not responsible for any errors or omissions. The information and data given in this user manual can be changed without prior notice. No part of this manual may be reproduced in any form or by any means without the written permission of the publisher, regardless of which method or which instruments, electronic or mechanical, are used. All technical information, drafts, etc. are liable to law of copyright protection.*

*Inclusion of Garmin copyrighted material in this presentation does not imply any endorsement by Garmin Ltd or its affiliates of the flight training material provided by Reality XP. Please note that Garmin Ltd. or its affiliates owns the copyright to this material and it is reproduced by permission.*