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.

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 (v1 to v6), Flight Simulator X - Steam Edition**, **Flight Simulator X** or **Flight Simulator 2004** for Windows[1].

---

[1] Versions required: P3D6, P3D5, P3D4, P3D3, P3D2, P3D1, FSX-SE, FSX-ACC/SP2, FS9-SP1; Windows versions required: Win7 to Win10

Getting Started

Installation

The add-on, which includes both[2] the GNS 530W and the GNS 430W simulation products, installs to[3]:

```
My Documents\Prepar3D v6 Add-Ons\Reality XP GNS\
```

During installation you will be prompted to download and install the Garmin GNS Trainer Lite[4]. 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\
```

Add-Ons Menu

The GNS 530W/430W V2 Menu loads with the simulator and is operated using the Prepar3D / FSim add-ons menu[5].

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.
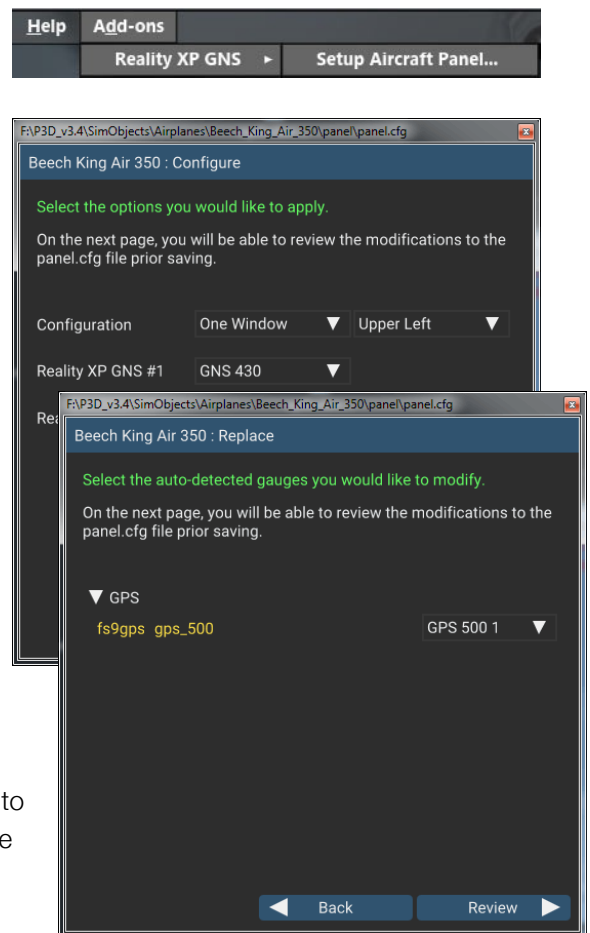
ⓘ The included Garmin trainer is a more recent version which embeds Main Software v3.3 released in 2010 and includes bug fixes, faster screen updates, uses less resources and adds new capabilities to the integrated navigator.

---

[2] The same gauge file installs both but It requires purchasing and installing both products to unlock both GNS models.

[3] P3D (v3 to v6): "Prepar3D v# Add-ons"; P3D2, P3D1, FSX-SE, FSX, FS9: in their respective "Gauges" folder.

[4] Compatible trainer versions: 3.3

[5] 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 Reality XP GNS offers a wide variety of configuration options in order to control the way each device operates and integrates in the aircraft. Most device configuration options are available in the configuration panel.

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

The panel saves the settings to the **RealityXP.GNS.ini** configuration file in the aircraft folder. When reloading the aircraft the add-on reads this file and restores the device settings.

🔒 **Lock and** 🛡 **Shield icons** indicate settings restrictions.
Read more about these in the *Third Party Vendor Settings* chapter.

🔁 **Reboot icon** shows when setting changes will reboot the device.

💡 **Drag and Drop** any RealityXP.GNS.ini file onto the panel window. All settings for the same device found in the dropped file will be imported.

ⓘ At the bottom of the options is a check box to **Optimize the Settings File**. It is designed to remove comments and leave only the necessary values to minimize file size. While this option is selected, the panel will overwrite the aircraft settings file after you make changes and click the Accept button. When the check box is cleared, the Accept button at the bottom of the panel changes to a Save button and all values you set in the panel are saved. This may result in a larger .ini file size.

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.

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.

When using a 'no bezel' gauge where the device screen covers the entire popup window surfaces, you can position the mouse cursor within a few pixels to the window edges in order to interact with the window resize handles or activate the window popup menu (right-click). In addition, the gauges can be manually configured to reserve and display a surrounding coloured fixed-size border for convenience.
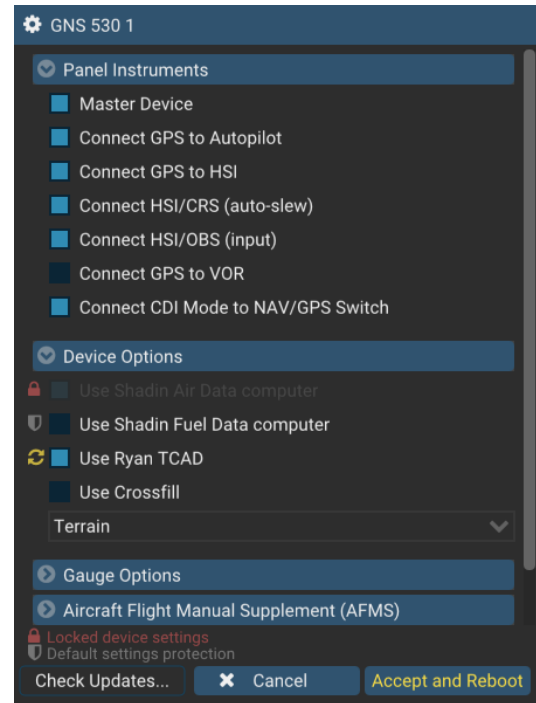
Features

Devices

The **GNS 530** add-on 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** add-on includes both the **GPS 400** and the **GNC 420**.

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).

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.

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

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

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 t**o 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)[6].
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.**

GPS Commands

The GNS may be configured to intercept all of the default FS GPS commands. Open the settings panel then select
"Use Simulator 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[7]. These are fully working replicas displaying valuable information on the panel and including a
convenient and functional **TERR INHB** button.
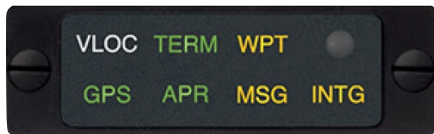
MD41-1484W Annunciator Control Unit   **MD41-1028 TAWS Control Unit**

---

[6] Systems and approaches availability depends on the navigation database.

[7] Ready-to-use with the pre-installed add-on.xml on Prepar3D (v4 to v6).
  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[8] 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.

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 **checklist.gns**[9] 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.

Hardware Controllers

The GNS supports a selection of external hardware controllers[10]. 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).

---

[8] 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.

[9] Supports both "checklist.txt" and "checklist.gns"

[10] 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.

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[11]:

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

Alternatively, this can 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.

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
```

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.

---
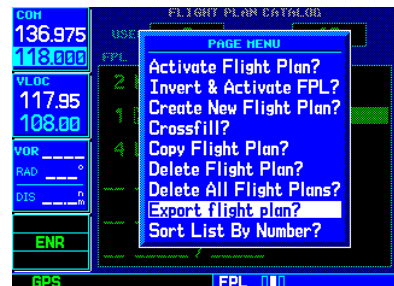
[11] The "FPL" folder must reside in the folder **GNSAPPDATA**. See Configuring Alternate Paths chapter for more information.

Settings

This chapter describes how to customize your GNS devices to the specific requirements of your aircraft. In settings you will be able to selectively link devices to panel instruments, configure devices capabilities, and adjust or configure other options[12]:

- Configure functionality and the user interface.
- Manage the settings which are specific to each aircraft.
- Modify each device settings independently.

Panel Instruments

Each GNS device has the ability to connect to different groups of panel instruments depending on the aircraft. The instruments connections can be added and removed depending on the aircraft panel gauges capabilities and the requirements of the user. Panel Instruments settings will configure the connections with the most common settings automatically.

Master Device	Each device connects to different instruments independently (example: GPS1 to VOR1 and GPS2 to VOR2). The Master Device has exclusive access when two or more devices, including the simulator GPS, are sharing the same resource (like the autopilot).
Connect GPS to Autopilot	Select if the device commands the flight director roll and pitch when in GPS mode. *Only if this is the "Master Device".*
Connect GPS to HSI	Select if the device overrides the HSI instrument (HSI simvars)
Connect HSI/CRS (auto-slew)	Select if the device auto-slew the HSI Course Selector when the DTK changes.
Connect HSI/OBS (input)	Select if the device uses the HSI OBS Selector for input when in OBS mode. *Only if the device is connected to HSI.*
Connect VOR#/OBS (input)	Select if the device uses the VOR OBS Selector for input when in OBS mode *Only if the device isn't connected to HSI.*
Connect GPS to VOR# indicator	Select if the device overrides the aircraft VOR indicator (NAV simvars). *See Gauge Options to select VOR1 or VOR2.*
Connect CDI Mode to NAV/GPS Switch	Select if the device CDI mode controls the aircraft NAV/GPS switch source. You can choose whether any or both the GPS and the Simulator has authority: `GPS → SIM, GPS ← SIM, GPS ⇄ SIM` *Set all RXP devices in the same aircraft to either GPS → SIM or GPS ⇄ SIM for best results.*

---

[12] **SHIFT+RIGHT Click the top edge of a GNS device** to open the configuration panel

Device Options

Each GNS has settings to allow customization of some of the device features. Each device embeds optional equipment which is used to augment navigation capabilities and pilot awareness. The most used settings are enabled by default.

Use Shadin Air Data computer	Simulate a Shadin Air Data computer by providing external air sensor data.
Use Shadin Fuel Data computer	Simulate a Shadin Fuel Data computer by providing external fuel sensor data.
Use Ryan TCAD	Simulate a Ryan 9900B TCAD (Traffic Collision Alerting Device) by providing surrounding traffic and collision alerts.
Use Crossfill	Automatically share route, waypoints and navigation data between two GNS *This must be enabled in both GNS devices used in the panel.*
TAWS Mode	Select which terrain alerting mode will be used: `Terrain Proximity, TAWS-B` *The alerting algorithm adapts the terrain alerting criteria based on nearby airports.*
TAWS Settings	The device must be configured to specify the minimum criteria that an airport must meet to be considered as a nearby airport for the purpose of TAWS/Terrain alerting: – Select the type of runway surface that the aircraft will typically use: `Hard, Hard & Soft, Water Only, Any` – Select or enter the shortest ground roll distance required for takeoff/landing. – Select or enter the minimum Approach, Flaps and Gear speeds (V_{REF}, V_{FE}, V_{LE}) *Auto-detected V speeds values will be saved in the settings file comments.*

Gauge Options

These settings adjust the way the plugin is integrating the GNS with the flight simulator and the aircraft panel. You can adjust options for a number of features that affect your in-cockpit integration and experience.

Radios	Select which radios the device synchronizes with: `COM1/NAV1, COM2/NAV2`
VLOC Source	Select which nav source the device connects to: `Auto, NAV1, NAV2` *This sets the VOR simvars the GNS will override.*
Fuel Type	Select which fuel type the fuel data sensor is probing: `AVGAS, JET-A, JET-B` *This sets the fuel density to convert weight to volume.*

Power Source	Select which power source the GPS is using:
	`Always On, Avionics On, General Panel On, Main Bus, Avionics Bus,`
	`Battery Bus, Hot Battery Bus, GenAlt #1 to #4`

Power Knob Switch	Select whether the GNS can switch off and how it is initially starting: `Volume Only, Default On, Default Off` *The power knob adjusts Prepar3D radios volume and will mute the radios when off.*

Clock Source	Select the GPS clock time source: `Simulator, Computer`
Display Settings	Define the device integral lighting options:
Screen	– Use the square box to select the mode: `Brightness, Rheostat`
Button	– Adjust the brightness (`0 to 100%`) or set the rheostat array dataref index *3rd party aircraft may provide rheostat buttons which are using the RXP rheostat simvars.*
Audio Settings	Define the device sound options:
Voice Type	Select the voice type: `Female, Male`
Master Volume	Adjust TAWS and GPWS volumes: `0 to 100%`
TCAD Volume	Adjust TCAD volume: `0 to 100%`
With Window Focus	Select whether the device play sounds only if P3D/FltSim window has the focus.
Advanced Settings	Define specific operational, input and output features:
Enable Failures	Select if the device supports failures *See the Failures Annunciation topic for more details.*
Use Simulator GPS Commands	Select if the device will use the P3D/FltSim GPS button and knob commands. *The device will intercept and use any GPS button which is typically used in 3D panels.*
Auto-Update Simulator GPS Waypoints	Select if the device overrides P3D/FltSim GPS legs with its own waypoints. *This may be used to display the device route waypoints into 3rd party EFIS.*
Auto-Save Active Route	Select if the device saves the active route to a file when it changes. *The file will be saved to: "My Documents\Flight Simulator Files\Reality XP GNS# Route.pln"*
Use Legacy Mouse Knobs	Select if the device will use legacy GNS mouse handling on knobs. *The knobs use colour coded mouse cursors and use left, right and middle mouse buttons.*
Use Legacy L:vars	Select if the device will publish legacy GNS simvars used for 3D cockpits.
HSI Deviation	Set the horizontal and vertical needle deflection scale. *This may be used to adjust the device output to 3rd party aircraft HSI/PFD gauges.*

Aircraft Flight Manual Supplement (AFMS)

The GNS supports enablement features, configured during installation in a real aircraft. Unless explicitly required you shouldn't modify any of these. You can find more detailed information in the official Garmin Installation Manuals.

CDI Key	`Disabled`: this causes the field above the CDI key to always display GPS, regardless of CDI key presses.
Selected Course for GPS	`Disabled`: ignore a selected course input for GPS operation in OBS mode. When OBS mode is selected, the course is entered on the controls of the GPS unit. *This setting is included for completeness only and has no practical use in P3D/FltSim.*
Selected Course for VLOC	`Disabled`: ignore the selected course input such that the VOR valid flag is only dependent on a valid VOR signal (lateral deviation calculated by another device). *This setting is included for completeness only and has no practical use in P3D/FltSim.*

V-Flag Declutter	Declutter: whenever vertical deviation is invalid, the vertical deviation bar is parked in the maximum UP position and the vertical flag is removed from view, except in the following cases: (i) the CDI is in VLOC mode and an ILS frequency is tuned, or (ii) the CDI is in GPS mode and a GPS approach with vertical guidance is active. In these cases, whenever the vertical deviation is invalid, the vertical deviation bar parks in the entered position and the vertical flag is shown.
GPS Selected	This will determine if the transition into approach mode is automatic or requires pilot acknowledgement of a message prompt. Prompt: when in GPS mode, the GPS Select discrete is active whenever an approach mode is active and the pilot has enabled the A/P APR Outputs (an associated message is displayed). This mode prevents the pilot to select automatic GPS to ILS CDI transitions on the AUX CDI/ALARMS page. *For Honeywell (Bendix/King) KFC 225 and KAP 140 using the GNS discrete signal.*
Engage Autopilot APPR	Enabled: automatically arm the autopilot approach mode when the A/P APR Outputs are active. *This is not a GNS device feature but a Reality XP option added for convenience..*
COM Presets	Enabled: the COM Remote Recall discrete input is connected in the installation. The user is able to preset COM frequencies and load them into the standby COM frequency using a remote switch. *This setting is reserved for future use.*

Other Settings

Use any text editor and manually add any of these in the **RealityXP.GNS.ini** file in the aircraft folder, in a **[GNS_###_#]** section.

SimGpsCmdsUnit =	Select the set of GPS commands the device will intercepts: 0 (auto), 1 (intercepts GPS#1 commands), 2 (intercepts GPS#2 commands)
NoHsiVarLimit = NoNavVarLimit =	False: limits HSI / NAV needles deviation to [-1,+1] (default) True: relaxes HSI / NAV needles simvars override limits
CpuAffinity = −1	Select the CPU which will run the Garmin trainer in the background: 1 to 32 (CPU#), 0 (last core), −1 (auto)
HardwareDevice =	Connect the GNS device to any of: OFF (none), PFC_STACK, PFC_430, PFC_530, #PID (ex: #D002 = PFC_430), or leave the value empty (first found) *Requires the Reality XP GNS Driver add-on or a professional license.*
HardwareIdx =	Select the hardware selector number or -1 to disable *Requires the Reality XP GNS Driver add-on or a professional license.*

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.

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** 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[13]:

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

In the example above, the 5th param was arbitrarily named 'NO_BEZEL'. 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]
```

Please note the following:

- The panel.cfg can only use one _1 and one _2 gauges at the same time (GTN or GNS V2).
- Without a 5th param a gauge uses the .DEFAULT section.
- With a 5th param the gauge uses the named section if any otherwise the default.
- With a 5th param set to NO_BEZEL, the gauge uses the .DEFAULT section but overrides the nobezel setting. This is a convenience shortcut saving you from having to create a section just for this.
- When you want to configure the settings for a gauge 'instance' in the panel.cfg file, you should create a named section and use the same name as the 5th parameter. Usually it is easier to start by copying the default section and change the .DEFAULT name/suffix.
- Sections and settings are optional. The gauge is always using default values.

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.

🛡 **How to configure 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.

🔒 **How to configure locked settings**: manually add 'locked' in a **[GNS_###_#]** section in the **RealityXP.GNS.ini** settings file. 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)

---

[13] Each line is in the following format: `gauge## = gauge_file!gauge_name, x, y, w, h, `*`param`*
ℹ Make sure the optional ***param*** is only using only letters, digits and underscore (no space),

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\FltSim\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: RNG UP	GPS_OUTR_CCW	rotate outer: GPS (ccw)
RUP	press button: RNG 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

Simulation 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[14]
rxp.gps.unit                     enum     _1 _2          4 chars encoded device id or 0[15]


Variable Name (RXP GNS V2 only)
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
rxp.radios.indicators.gns_version enum                   version num (ex: v2.5.10.0 = 0x02050A00)
```

Control Variables

Third party add-ons can interact with the following variables to further enhance GNS integration with the aircraft:

```
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)
```

Please note: third party add-ons should only write 1 to any "rxp.gps.dscrt_in.xxxx" in order to trigger the input signal. The GNS reverts the value to 0 once the signal is consumed.

---

[14] 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;`

[15] **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[16]:

```
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
```

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).
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.
```

---

[16] 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, them 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[17] 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)[18]:

```
    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"
```

---

[17] how to edit environment variables: https://superuser.com/questions/1334129/setting-an-environment-variable-in-windows-10-gpodder

[18] 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):

```
GNSADBFILE="C:\Users\MyName\Documents\gns_db.bin"          (alias: GNSDBFILE)
GNSODBFILE="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.

Gauges Dimensions

For best legibility we recommend using gauge dimensions which are integral multiples of their native sizes when configuring aircraft panels:

*   5XXX with bezel: 980 x 714 (displays the screen at 2:1 ratio, texture is 980x714)
*   4XXX with bezel: 980 x 412 (displays the screen at 2:1 ratio, texture is 980x412)
*   5XX screen only: 320 x 234 (native screen resolution)
*   4XX screen only: 240 x 128 (native screen resolution)

Integration Guidelines

This chapter describes some of the best practices to configuring and distributing settings files, for third party aircraft vendors and aircraft mod enthusiasts. Using the 'Third Party Vendor Settings' options, settings can be configured in a way which enforces both default values and prevents accidental end-user changes which may alter the intended integration with the aircraft.

Normal, Defaults and Locked Settings

Settings in the main **RealityXP.GNS.ini** file in the aircraft folder follow these conventions:

- Some settings are used for the integration with the aircraft and the panel (ex: LinkHsi).
- Some settings are used for the presentation (ex: nobezel).
- Others depend on the end user preferences (ex: popleft).
- Sections and settings not found in the main ini file are always replaced internally by a default value.

Settings in the optional **RealityXP.GNS.defaults.ini** file in the aircraft folder follow these conventions:

- They override the main ini file ever time the device reloads.
- They have precedence regardless whether they exist in the main settings file.
- They display a grey shield icon in the GUI Configuration Panel.
- Locked settings can't be changed in the GUI Configuration Panel and display a red lock icon.

Minimal Settings

The smallest ini file for configuring an aircraft with a GNS 530 Unit# is empty. If there is no RealityXP.GNS.ini file in the aircraft folder, all the settings are set to their defaults, including the missing [GNS_530_1] section.

Please note:

- Let the plugin create a new ini file from scratch in order to find out what each default settings values are.
- Make sure to opt-out the 'Optimize Settings' option in the GUI Configuration Panel in order to let the plugin save all available setting option with their default values.
- "locked" is supported since initial release.
- defaults.ini file is supported since GTN v2.5.11 and GNS V2 v2.4.6
- "FuelType = AUTO" is supported since GTN v2.5.14 and GNS V2 v2.4.9

Practical Example

Let's try to setup the setting files needed for a typical aircraft. We'll lock some of the options and their values which are specific to this aircraft panel instruments, and set default rendering options which will ensure the expected end-user experience regardless of any changes to the main ini file.

RealityXP.GNS.defaults.ini

```
[GNS_530_1]
locked = LinkHsi, LinkCrs, LinkObs, LinkVor, AutoNavGps, FuelType, UseSimGpsCmds
LinkHsi = true
LinkCrs = true
LinkObs = true
LinkVor = true
AutoNavGps = true
FuelType = AUTO
UseSimGpsCmds = false
GsiScale = 0.8

[GNS_530_1.DEFAULT]
nobezel = false
usemouse = true
popleft =
popright = 15531

[GNS_530_1.PANEL3D]
nobezel = true
usemouse = true
popleft =
popright = 15531
```

The aircraft panel.cfg has 2 instances of the GNS 530 #1, one in a popup window with the ident 15531 and another instance in the virtual cockpit using 'PANEL3D' as the 5th param in order to use specific rendering settings for the VC.

The [GNS_530_1] section in the 'defaults' file overrides the section with the same name in the main settings file. Its options are set to the designed values for this aircraft and they also revert any manual change in the main ini file when loading the aircraft. They are further locked in order to prevent any user change in the GUI.

Similarly, both the [GNS_530_1.DEFAULT] and the [GNS_530_1.PANEL3D] sections provide default values and guarantee the device will always render as expected even if the main ini file is corrupted or deleted. There is no locked= setting for rendering sections because there is no GUI to change them.

The GsiScale setting has a default value but is not locked. This means the end user can change it during flight but it will still revert to the default value anytime the user reloads the aircraft.

RealityXP.GNS.ini

With most options set from the default.ini file, the main ini file setting file only requires the few values necessary for the first time the user loads the aircraft:

```
[GNS_530_1]
; is the master device if true.
MasterDevice = true
```

Troubleshooting

This chapter includes solutions 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\FSX\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

P3D6 / P3D5 / P3D4: open notepad, new text document, copy/paste the following content, save the file as[19]:

My Documents**Prepar3D v6 Add-ons**\\Reality XP GNS\\add-on.xml

```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>
```

---

[19] Use the folder corresponding to the Prepar3D version: **"v6", "v5" or "v4"**. 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[20]:

```xml
<?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. Use the add-on menu **"Open Log Files…"** to open and review the content of both log files and to start troubleshooting.

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: enable Garmin Trainer logs. 4: enable Trainer version and paths logs. 5: enable all the above (1+4 = 5)
`DebugValue = 0`	2: render power-off screen in red.

---

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

Miscellaneous

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 it 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.