RXP GTN 750/650 TOUCH

User's Manual For Prepar3d / Flight Sim

Tap into the Future of Avionics!

With GTN 650/750's revolutionary touchscreen interface, you're rarely more than 2 taps away from all of the system's primary functions! The intuitive touchscreen controls and large display give you unprecedented access to high-resolution terrain mapping, graphical flight planning, geo-referenced charting, advanced navigation and more.

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 GTN 750 and GTN 650 Touch 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.

Graphical flight planning capabilities allow you to preview your entire route on the map screen and easily enter new waypoints or modify existing ones. A handy "rubber band" feature lets you grab any leg of your flight plan route and move it to accommodate a deviation to your flight plan.

With the GTN, you can set your standby Com frequency simply by touching the onscreen number pad. Another touch of the screen flip-flops your active and standby frequencies. Recent and favourite user frequencies also are a breeze to access. The GTN also automatically decodes a station's Morse code to provide a positive identification – no aural decoding required.

The GTN can control the simulator audio panel (GTN 750 only) as well as the transponder, with optional IDENT and REPLY annunciations. This setup could save you up to 3-inch of panel space versus a typical GNS 430/530 install. In addition, precise course deviation and roll steering outputs can be coupled to the simulator autopilot so that IFR flight procedures may be flown automatically. This includes all ARINC 424 leg types, including NextGen radius-to-fix (RF) legs.

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 GTN 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 GTN 750/650 Touch is an add-on for **Prepar3d v4/3/2/1, Flight Simulator X - Steam Edition**, **Flight Simulator X** or **Flight Simulator 2004** for Windows[1].

---

[1] 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[2] the GTN 750 and the GTN 650 simulation installs to[3]:

```
My Documents\Prepar3d v4 Add-ons\Reality XP GTN\
```

During installation you will be prompted to download and install the Garmin GTN/GDU Trainer[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\GtnTrainer\
```

Add-ons Menu

The GTN 750/650 Touch Menu loads with the simulator and is operated using the Prepar3d / FSim add-ons menu[5].

Select the **menu Add-ons | Reality XP GTN** to add or remove GTN 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 GTN popup windows.
- Replace any similar GPS gauge with a GTN gauge using an auto-detect feature.
- Remove all GTN 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.

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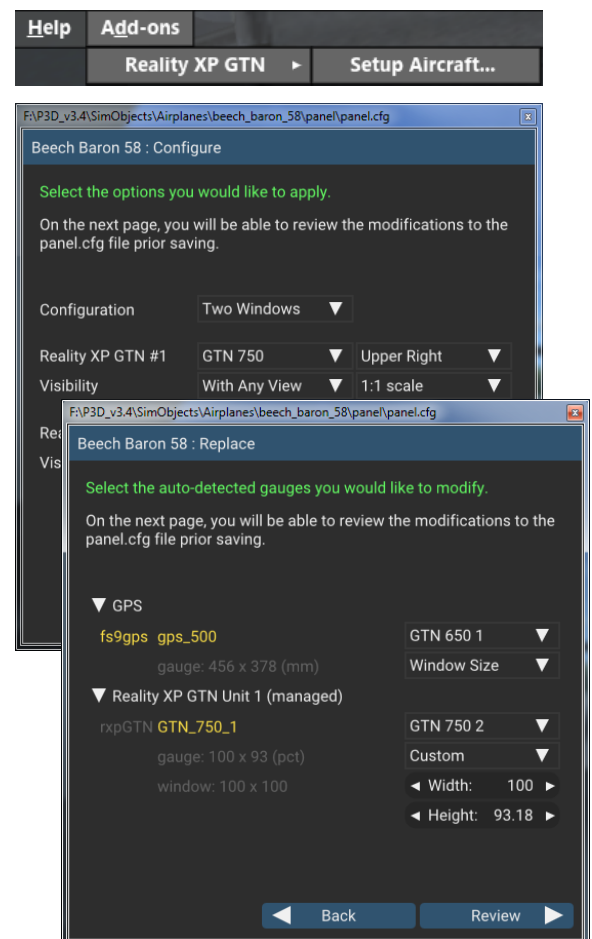
[2] The same gauge file installs both but It requires purchasing and installing both products to unlock both GTN models.

[3] P3D3: "Prepar3d v3 Add-ons"; P3D2, P3D1, FSX-SE, FSX, FS9: in their respective "Gauges" folder.

[4] Compatible trainer versions: v6.62, v6.50.x, v6.21

[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 GTN 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 GTN device to open the configuration panel.

The panel saves the settings to the **RealityXP.GTN.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 settings changes will reboot the device.

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

ℹ️ 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.
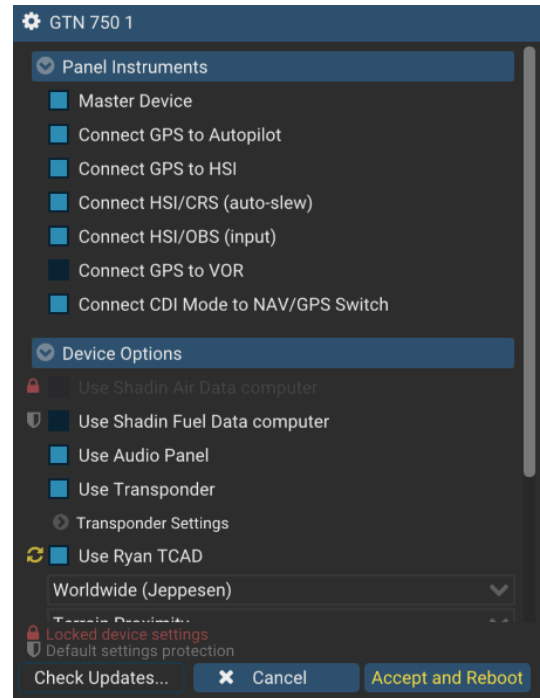
Using the Mouse

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.
- Touchscreen actions are carried naturally with single clicks or click-and-drag. Mouse wheel simulates pinch-in and pinch-out actions.

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 GTN device to scale the window to either 1:1 or 2:1 screen sizes.

Features

Devices

The **GTN 750** plugin includes the **GTN 725**, similar in form and function to the 750, it is a radio-less sibling and fits panels already equipped with radios. Similarly, the **GTN 650** plugin includes both the **GTN 635** and the **GTN 625**.

Radios

The GTN 750 and the GTN 650 can control the simulator COM and NAV audio panel volumes using the **Volume** knob (P3D2.5+ only).

Autopilot

The GTN 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 GTN embeds a **Ryan 9900BX 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 GTN touchscreen. The Flight Simulator AI Traffic simulation variables feed the TCAD simulation.

TAWS

The GTN offers **TAWS-A, TAWS-B, HTAWS and TERR Proximity modes** with advanced settings available to fine tune the capabilities depending on the aircraft configuration. When configured to operate in TAWS-A mode, the GTN gauge automatically detects best V_{REF}, V_{FE} and V_{LE}. The GTN 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.

Additional Features

The GTN offers some nice additional features such as setting the **touchscreen 'click' sound volume** or adjusting the **screen backlight level** directly on the unit. Please refer to the 'Garmin GTN Cockpit Reference Guide'.

Discrete Inputs

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

GPS Commands

The GTN 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. The GTN intercepts the following Flight Simulator GPS bezel buttons and knobs commands:

- GPS CRS knobs control the GTN FMS knobs (lower right).
- GPS BUTTON 1 (push), 2 (turn left) and 3 (turn right) controls the GTN Volume knob (upper left).
- DTO key controls the GTN DTO key.
- MENU key controls the GTN HOME key.
- NRST controls CDI mode, OBS controls OBS mode, TERRAIN controls TAWS inhibit.
- Other GPS keys are disabled.

Keyboard Shortcuts

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

Additional Gauges

In addition to the main GTN gauges, the product installs additional **Mid-Continent gauges** ready to be added into any panel[6]. These are fully working replicas displaying valuable information on the panel and including a convenient and functional **TERR INHB** button.
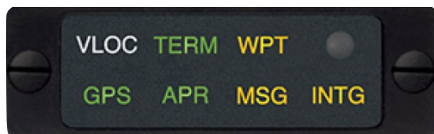
MD41-1484W Annunciator Control Unit

MD41-1028 TAWS Control Unit

---

[6] 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 GTN plugin uses the following sim variables[7] 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.	
PARTIAL PANEL COMM:#	COM radios failure only.	
PARTIAL PANEL NAV:#	NAV radios failure only.	
PARTIAL PANEL TRANSPONDER	Transponder failure.	
CIRCUIT MARKER BEACON ON	Audio Panel failure.	

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. In the Checklist Menu, you can access any Checklist, or group of Checklists, and clear the current or all Checklists.

Checklists are created using the Garmin Checklist Editor software[8] and must be saved to a file named **checklist.gtn**[9] into the aircraft folder. To get started, a sample pack installs to the following folder:

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

Unzip the "checklists.zip" archive file to extract the sample ".ace" checklists files, then copy one of these to the aircraft folder and rename the file to checklist.gtn. On the GTN device, while viewing the Utilities page group, touch the **Checklists** key to start the Checklists function. Touch the **Menu** key to select an option from the Menu.

---

[7] 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.

[8] Available online: https://www8.garmin.com/support/download_details.jsp?id=5075

[9] Supports both "checklist.ace" and "checklist.gtn"

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 GTN. The imported flight plans can then be previewed, used directly or stored to the GTN flight plan catalog. This chapter describes how to enable and use this feature[10].

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 **FPLN** and be located in the GTN Trainer Data folder, typically[11]:

```
"C:\ProgramData\Garmin\Trainers\Databases\FPLN"
```

Alternatively, this can be a "FPLN" shortcut linked to any other folder on the hard drive, for example:

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

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

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

Importing flight plans

The last step is to import the flight plans and the user waypoints into the GTN device. To proceed with the starter pack, unzip the "flightplans.zip" archive file to extract the "KSLE_KSLE.gfp" flight plan and the "user.wpt" sample files, then copy the two files to the "FPLN" folder.

On the GTN device, while viewing the Flight Plan page, touch the **Menu** key and then the **Catalog** key to display the Flight Plan Catalog.
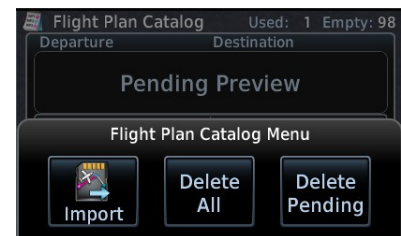
An Import key will be present in the Catalog menu bar when flight plans are present on the virtual datacard folder. Touch the **Catalog Menu** key, then the **Import** key to open a pop-up with a list of the file names of the flight plans on the datacard, then select the desired flight plan to import.

Importing user waypoints

The GTN can import user generated waypoints from a file on the datacard. The created waypoints will be at the latitude and longitude specified in the file with the specified name and comment.

When a "user.wpt" user waypoint file is present in the virtual datacard FPLN folder, a key will be available on the Waypoint Info page for importing user waypoints. From the Waypoint Info page, touch the **Import Waypoints** key. Touch **OK** to acknowledge the pop-up to import all of the user waypoints in the file.

---

[10] See "GTN Flight Plan and User Waypoint transfer.pdf" guidelines for how to create text based flight plan files (included in the starter pack).

[11] The **FPLN** folder must reside in the folder known as **GTNNAVDATA** (GTN Trainer 6.50+) or **GTNAPPDATA** (previous versions).
See Configuring Alternate Paths chapter for the meaning and location of the trainer known folder names.
Trainer 6.21: "C:\ProgramData\Garmin\GTN Trainer Data\GTN\FPLN".

Settings

This chapter describes how to customize your GTN 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 will take exclusive access when two or more devices are sharing the same connection (like the autopilot).
Connect GPS to Autopilot	Select if the device commands the flight director roll and pitch when in GPS mode.
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.

Device Options

Each GTN 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.

---

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

Use Shadin Fuel Data computer	Simulate a Shadin Fuel Data computer by providing external fuel sensor data.
Use Audio Panel	Simulate a GMA 35 Audio Panel and enable touchscreen audio panel interface.
Use Transponder	Simulate a GTX 327 Transponder and enable touchscreen interface. *Additional settings may appear when changing mode.*
Transponder Settings	Select which VFR code will be used: `User Defined, 1200 (US), 7000 (EUR)` *User defined code can be manually set; see "Other Settings" below.*
Use Ryan TCAD	Simulate a Ryan 9900BX TCAD (Traffic Collision Alerting Device) by providing surrounding traffic and collision alerts.
Database Type	Select which database file included with the Garmin GTN trainer will be used: `Worldwide, North America, Americas Rotor (Jepp), North Am. Rotor (Grmn)`
TAWS Mode	Select which terrain alerting mode will be used: `Terrain Proximity, TAWS-B, TAWS-A` *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 GTN 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 GTN will override.*
Fuel Type	Select which fuel type the fuel data sensor is probing: `AVGAS, JET-A, JET-B, DIESEL` *This sets the fuel density to convert weight to volume.*
Ownship Icon	Select which aircraft symbol will display on the map page. *The "Auto" mode selects the symbol using the number of engines and the aircraft type.*
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`

Clock Source	Select the GPS clock time source: `Simulator, Computer`
Display Settings	Define the device integral lighting options:
Screen Button Use Half-Rate Screen Update	– Use the square box to select the mode: `Brightness, Rheostat` – Adjust the brightness (`0 to 100%`) or set the rheostat array dataref index – Select whether the screen draws every other frame to preserve resources. *3rd party aircraft may provide rheostat buttons which are using the RXP rheostat simvars.*
Audio Settings	Define the device sound options:
Voice Type Master Volume TCAD Volume With Window Focus	Select the voice type: `Female, Male` Adjust TAWS and GPWS volumes: `0 to 100%` Adjust TCAD volume: `0 to 100%` 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 GTN# Route.pln"*
Use Legacy Mouse Knobs	Select if the device will use legacy GTN 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 GTN 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 GTN 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..*
Airspace Labels	`Enabled`: The Airspace Label feature shows the United States airspace system altitude limits up to a selected range.
Vertical Navigation Type	Select which vertical navigation assistance GPS is using: `VCALC, VNAV` *VNAV is a new vertical navigation feature introduced with GTN software v6.50.*
Transition To Approach (VNAV)	− `Enabled`: attempts a smooth transition from en route to approach vertical guidance. Aircraft intercepts with approach guidance from below the glidepath. − `Disabled`: en route VNAV terminates at the waypoint prior to the FAF on approaches with vertical guidance, or at the FAF (LNAV only).
RF Procedure Legs	`Enabled`: fly Instrument Approach Procedures (IAPs) with radius-to-fix (RF) legs.
Mark On Target	`Enabled`: additional user interface for handling mark on target waypoints will be present. *This setting enables the MARK ON TRGET command.*

Other Settings

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

`CpuAffinity = −1`	Select the CPU which will run the Garmin trainer in the background: `1 to 32 (CPU#), 0 (last core), −1 (auto)`
`VFRCode = #1200`	Select the transponder code for VFR.

Keyboard Shortcuts

This chapter describes the list of custom commands available to control the GTN 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".

Open the file "My Documents\Reality XP\GTN Simulation\FltSim\RealityXP.GTN.ini" with notepad to configure the hotkeys. 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.

DTO	press button: Direct To
HOME	press button: Home
VOL_PUSH	press knob: VOL
VOL_CCW	rotate knob: VOL (counter clock-wise)
VOL_CW	rotate knob: VOL (clock-wise)
FMS_PUSH	press knob: FMS
FMS_OUTER_CCW	rotate knob: FMS (counter clock-wise)
FMS_OUTER_CW	rotate knob: FMS (clock-wise)
FMS_INNER_CCW	rotate knob: FMS (counter clock-wise)
FMS_INNER_CW	rotate knob: FMS (clock-wise)
CDI_SOURCE	toggle CDI source (discrete input)
OBS_MODE	toggle OBS mode (discrete input)
TAWS_INHIBIT	inhibit TAWS (discrete input)
MARK_ON_TARGET	mark current position (discrete input - requires AFMS Mark On Target Enabled)
TOGGLE_WINDOW	toggle window visibility
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

Configuration

Aircraft Configuration

This chapter details the general organization of the settings file. The GTN settings are automatically saved in the file **RealityXP.GTN.ini** located in the aircraft folder. Every time a GTN opens from the Reality XP GTN 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.

```
[GTN_750_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

[GTN_750_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) (disables touchscreen when set).
popleft =
```

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

- The **GTN_750_1** gauge section contains the settings for this device.
- The **GTN_750_1.DEFAULT** section contains the default settings or 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 GTN 750 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.GTN.ini** supports per-instance configuration.

The automatically generated RealityXP.GTN.ini file in the aircraft folder has **[GTN_###_#.DEFAULT]** sections for each device found in the panel. Each gauge sibling uses the default settings from this section, for example [GTN_750_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.GTN.ini file, change the suffix from 'DEFAULT' to 'NO_BEZEL' then edit the new section settings.

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

```
[GTN_750_1.NO_BEZEL]
; show screen only gauge if true.
nobezel = true
...
```

The **panel.cfg** file may be configured like this[13]:

```
[VCockpit01]
gauge00=rxpGTN!GTN_750_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:

```
[GTN_###_#.MY_NICE_VC]
[GTN_###_#.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 red lock icon in the GUI settings panel. Default settings appear with a grey shield icon and reset to their default value every time the device reloads.

⛊ **How to configure default settings**: create a **RealityXP.GTN.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 **[GTN_###_#]** section in the **RealityXP.GTN.ini** settings file. For example, an aircraft requires the device to be set to 'Copilot' side 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`*

   ⓘ Make sure the optional *param* is only using only letters, digits and underscore (no space),

Simulation Variables

This chapter describes the list of additional sim variables the GTN 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 GTN only)
rxp.radios.indicators.gtn_unit1    enum                  4 chars encoded device id or 0
rxp.radios.indicators.gtn_unit2    enum                  4 chars encoded device id or 0
rxp.radios.indicators.gtn_version  enum                  version num (ex: v2.5.10.0 = 0x02050A00)
```

---

[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)

Control Variables

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

```
rxp.panel.rheostat_screen               number        (0 to 100)
rxp.panel.rheostat_integral             number        (0 to 100)
rxp.transponder.mode                    enum          (0:STBY, 1:ON, 2:GRND, 3:ALT)
rxp.transponder.light                   enum          (0:AUTO, 1:'R' OFF, 2:'R' ON)
rxp.transponder.id                      boolean       (squawking if true)
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 (write 1 to trigger)
rxp.gps.dscrt_in.mark_on_target_1(_2)   boolean       discrete in signal (write 1 to trigger)
```

rxp.panel.rheostat_screen rxp.panel.rheostat_integral	Third party add-on can use these variables to control the GTN screen and keys integral lighting independently, with their panel light rheostats if any.
rxp.transponder.mode	Third party add-on can use this variable and synchronize the GTN transponder mode: • When the variable value changes then update the other add-on with the new mode. • When the other add-on mode changes then write the new value to the variable and the GTN transponder mode will change.
rxp.transponder.light	Third party add-on can control the reply state indicated with an 'R' flag: • Write 0 to let the GTN simulate a realistic 'R' flag (default). • Write 1 to turn the reply flag off. • Write 2 to turn the reply flag on.
rxp.transponder.id	Third party add-on can write 1 or 0 to directly change the GTN transponder IDENT mode. Otherwise when the user is squawking the GTN transponder, the value changes to 1 for a few seconds and reverts to 0 automatically.
rxp.gps.dscrt_in.xxxx	Third party add-on can write 1 to trigger an input signal. The GTN reverts the value to 0.

GPS Commands

The GTN 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_DIRECTTO_BUTTON         GTN Direct To.
GPS_MENU_BUTTON             GTN Home.
GPS_CURSOR_BUTTON           GTN FMS push.
GPS_GROUP_KNOB_INC          GTN FMS chapter up (right outer knob cw).
GPS_GROUP_KNOB_DEC          GTN FMS chapter dn (right outer knob ccw).
GPS_PAGE_KNOB_INC           GTN FMS page up (right inner knob cw).
GPS_PAGE_KNOB_DEC           GTN FMS page dn (right inner knob ccw).
GPS_BUTTON1                 GTN VOL push.
GPS_BUTTON2                 GTN VOL up (left knob cw).
GPS_BUTTON3                 GTN VOL dn (left knob ccw).
GPS_NEAREST_BUTTON          GTN CDI MODE (discrete in).
GPS_OBS_BUTTON              GTN OBS MODE (discrete in).
GPS_TERRAIN_BUTTON          GTN TAWS Inhibit (discrete in).
GPS_CLEAR_ALL_BUTTON        GTN Discrete In (uses command modifiers, 15: Mark On Target).
```

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 GTN uses this existing parameter in order to modify the "KEY_GPS" (C++) or the "GPS" (XML) command action. This enhances integration between the GTN 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 **HOME** key **press-release** event to **GPS#1**: `GPS_MENU_BUTTON, 0`
Sending the **HOME** key **press-release** event to **GPS#2**: `GPS_MENU_BUTTON, 8`   = *bit4 (8)*
Sending the **HOME** key **press and hold** event to **GPS#2**: `GPS_MENU_BUTTON,10`   = *bit2 (2)* + *bit4 (8)*
Sending the **HOME** key **release** event to **GPS#2**: `GPS_MENU_BUTTON,12`   = *bit3 (4)* + *bit4 (8)*

ModelDef.xml Example

A complete canonical implementation installs to:
`C:\Program Files (x86)\Reality XP\GTN Simulation\FltSim\modeldef (RealityXP GTN).xml`

Here is how to implement a typical **<PartInfo>** for the GTN Device #2 "Home" button:

```
  <Name>RXP_GTN_HOME_BUTTON_2</Name>
  <MouseRect>
    <Cursor>Hand</Cursor>
    <TooltipText>Home / Map (Press and Hold)</TooltipText>
    <MouseFlags>LeftSingle+LeftRelease+DownRepeat</MouseFlags>
    <CallbackCode>
      (M:Event) 'LeftSingle' scmp 0 == if{ 10 (>K:GPS_MENU_BUTTON) }
      (M:Event) 'LeftRelease' scmp 0 == if{ 12 (>K:GPS_MENU_BUTTON) }
    </CallbackCode>
  </MouseRect>
```

Device Embedded Memory

The GTN 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\Trainers\GTN\nonvol              (GTN Trainer 6.51+)
C:\ProgramData\Garmin\GTN Trainer Data\GTN\GTN1nonvol  (GTN Trainer 6.21 and below)
```

The Reality XP GTN 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: GTN 625 #1 and GTN 625 #2
sys_nandc, sys_nandd: GTN 635 #1 and GTN 635 #2
sys_nande, sys_nandf: GTN 650 #1 and GTN 650 #2
sys_nandg, sys_nandh: GTN 725 #1 and GTN 725 #2
sys_nandi, sys_nandj: GTN 750 #1 and GTN 750 #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 GTN 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[16] to configure alternate trainer paths in a typical scenario.

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

```
GTNSIMPATH="C:\Program Files (x86)\Garmin\Trainers\Packages\GTN\bin"    (Trainer v6.50+)
GTNSIMPATH="C:\Program Files (x86)\Garmin\GTN Trainer\GTN\sys\sim"      (older Trainers)
```

Set parent path to trainer data files (Resources, Audio, etc..):

```
GTNAPPDATA="C:\Program Files (x86)\Garmin\Trainers\Packages\GTN\data"  (Trainer v6.50+)
GTNAPPDATA="C:\ProgramData\Garmin\GTN Trainer Data\GTN"                (older Trainers)
```

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

```
GTNNAVDATA="C:\ProgramData\Garmin\Trainers\Database"                   (Trainer v6.50+)
GTNNAVDATA="C:\ProgramData\Garmin\GTN Trainer Data\DB"                 (older Trainers)
```

---

[16] how to add environment variables:
http://www.kunal-chowdhury.com/2016/09/windows-10-anniversary-update-how-to-edit-environment-variables.html

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 gtn_db.bin):

```
GTNADBFILE="C:\Users\MyName\Documents\gtn_db.bin"          (alias: GTNDBFILE)
GTNODBFILE="C:\Users\MyName\Documents\terrain.odb2"        (alias: GTNOBSTCL)
```

Installing Multiple Garmin Trainers

This chapter describes a procedure when you want or need to install two different versions of the Garmin GTN Trainer on your computer. The Garmin GTN Trainer installer prevents installing two different versions, such as the latest compatible with the Reality XP GTN Touch and an older one. However, the GTN Touch 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 GTN Touch

2. Copy the "sim" folder located in "C:\Program Files (x86)\Garmin\GTN Trainer\GTN\sys\" to the location of your choice[17], or rename it for example to: "C:\Program Files (x86)\Garmin\GTN Trainer\GTN\sys\sim (6210)\"

3. Copy the "GTN Trainer Data" folder located in "C:\ProgramData\Garmin\" to another location, or just rename it to something like: "C:\ProgramData\Garmin\GTN Trainer Data (6210)"

4. Add the following 2 environment variables in Windows:
   name = GTNSIMPATH value = *the path to the 'sim' folder you have chosen above*
   name = GTNAPPDATA value = *the path to the 'data' folder you have chosen above*
   for example:
```
GTNSIMPATH="C:\Program Files (x86)\Garmin\GTN Trainer\GTN\sys\sim (6210)\"
GTNAPPDATA="C:\ProgramData\Garmin\GTN Trainer Data (6210)"
```

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

---

[17] Please note you only need to copy or rename this folder for the GTN plugin, but you'll need the rest for our G500/600

Troubleshooting

This chapter describes 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

P3D4: open notepad, create a new text document, copy/paste the following content, then save the file as[18]:

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

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

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

---

[18] 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[19]:

```
<?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 GTN Menu</Name>
        <Disabled>False</Disabled>
        <ManualLoad>False</ManualLoad>
        <Path>C:\Program Files (x86)\Reality XP\GTN Simulation\FltSim\32\rxpGTN_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\rxpGTN.gau.log**".
The GTN Simulation log file: "**My Document\rxpGtnSim.dll.log**"
The add-on menu log file: "**My Documents\rxpGTN_menu.dll.log**"

Debug Settings

The GTN 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 "rxpGtnSim.dll.log"

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

`LogTrainer = 0`	1: enable Garmin Trainer logs. 2: enable Garmin Trainer crash report handler (trainer 6.50+ only). 4: enable Trainer version and paths logs. 7: enable all the above (1+2+4 = 7)
`DebugValue = 0`	1: enable Video Card Driver debug logs. 2: render power-off screen in red.

---

[19] 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.

Intel integrated Graphics vs Discrete Graphics card

The Garmin trainer "GTN Simulator.exe" process must use discrete graphics instead of the integrated otherwise the Intel drivers exhibit bugs and/or black screen symptoms (systems with Intel graphics only should run fine).

Force App To Use NVIDIA Graphics Card

• Open the NVIDIA control panel. Right-click on an empty area on your desktop and select the 'NVIDIA Control Panel' option from the context menu.
• Go to the Desktop menu and enable the 'Add "Run with graphics processor" to Context Menu' option.
• Return to your desktop. Right-click the app you want to force to use the dedicated GPU. The right-click context menu will have a 'Run with graphics processor' option. Select 'High-performance NVIDIA processor' from the sub-options and the app will run using your dedicated GPU.

Force App To Use AMD Graphics Card

• Open the AMD Catalyst Control Center app by right-clicking on an empty area on your desktop. Select 'Catalyst Control Center' from the context menu. From the list of panels on the left, select 'Power'. Under 'Power', select 'Switchable application graphics settings'.
• Here you will see a list of apps that AMD already recognizes as needing better GPU support. On this panel is an 'Add Application' button. Click it and select the EXE of the app you want to force to use the AMD graphics card.
• After you add the application, open the dropdown next to it for setting graphics options. Select 'High performance' and you're done. The app will always use AMD graphics card.