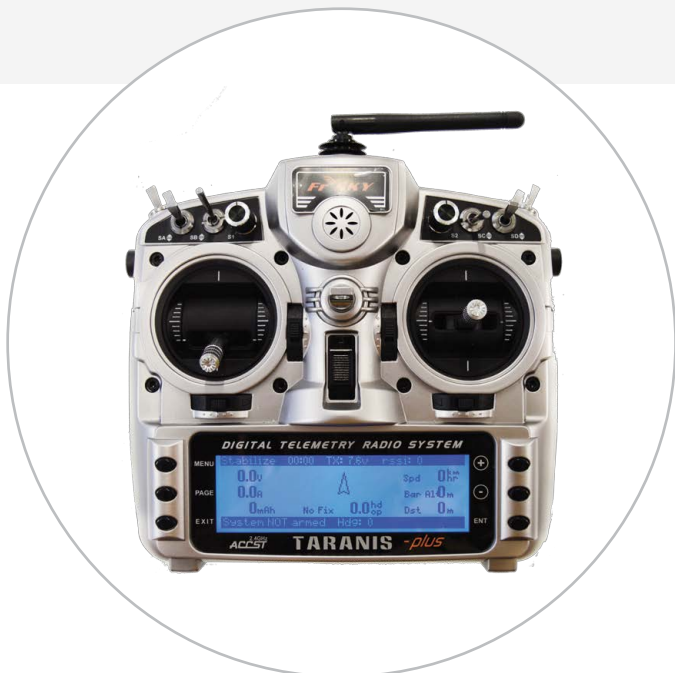




Airborne Projects specializes in building drone solutions with emphasis on telemetry gathering and integration with avionics and automatic flight systems.

MAVLink to FrSky Smart Port (S.Port) Converter

Quick Start Guide



Airborne Projects' MAVLink to FrSky Smart Port Converter is a telemetry relay board that sits between the telemetry port of your APM, Pixhawk or NAVIO+ autopilot module and the Smart Port of your FrSky X8R receiver sending your flight controller telemetry to your Taranis radio.

Through in-house development we are happy to present you a cheaper and open source friendlier alternative to the Teensy board commonly sold for this purpose.

Contents

- 1 Context** **1**

- 2 Getting Started** **1**

- 3 Assumptions** **1**

- 4 Configuring the Telemetry on Mission Planner** **1**
 - 4.1 Using TELEM 1 or TELEM 2 1
 - 4.2 Using SERIAL 4/5 Port 2

- 5 Loading the LUA telemetry scripts to the Taranis X9D transmitter** **2**

- 6 Troubleshooting** **3**

- 7 Further Information** **3**

- Appendix A How to update the Converter's firmware** **4**

- Appendix B GUI Description** **5**

- Appendix C Product Description** **6**

- Appendix D Specifications** **6**

1 Context

This manual's goal is to get the Airborne Projects' **Mavlink to FrSky Smart Port Converter** working with OpenTX 2.1.X, where X is a version number higher or equal to 3.

The Pixhawk/Navio+ version's firmware can be upgraded via USB, using 'hex' binaries provided by Airborne Projects.

2 Getting Started

Mavlink to FrSky Smart Port Converter is a telemetry relay board that sits between the telemetry port of your APM 2.X, Pixhawk on NAVIO+ autopilot modules and the Smart Port of your FrSky X8R receiver sending the autopilot's telemetry to your Taranis X9D radio.

3 Assumptions

1. The Taranis X9D radio transmitter is bound in **D16 mode** to its X8R receiver which is connected to the **MavLink to FrSky SmartPort Converter** through its Smart Port. It is very important the correct selection of the binding mode as other modes may not have telemetry. Check <https://www.youtube.com/watch?v=1IYg5mQdLVI> for details on how to do it.
2. The **MavLink to FrSky SmartPort Converter** module is connected to the APM/Pixhawk/NAVIO+ telemetry output;
3. The power to the **MavLink to FrSky SmartPort Converter** is supplied by the autopilot's telemetry port;
4. The **MavLink to FrSky SmartPort Converter** has the firmware originally loaded or recommended by Airborne Projects;
5. You have downloaded the zip file containing the HEX file and the LUA scripts from **here**;

Note: On APM boards the available telemetry port should be used; On NAVIO+ autopilot the interface is the UART port and for Pixhawk users all the **Telemetry 1,2,4/5** ports can be used.

4 Configuring the Telemetry on Mission Planner

4.1 Using TELEM 1 or TELEM 2

1. Connect you flight controller to Mission Planner (or equivalent);
2. Select the **Config/Tuning** tab;
3. Select **Full Parameter Tree**;
4. Find the command node **SR1**¹ and set:
 - (a) SR1_EXT_STAT 10
 - (b) SR1_EXTRA1 0

¹For further information about this parameters please visit **ArduCopter's SR1 official documentation**. SR1 or SERIAL1 is a placeholder for any of the telemetry channels and can be taken as SR(a) or SERIAL(a), where (a) is the number of the telemetry port where the converter is connected.

- (c) SR1_EXTRA2 10
 - (d) SR1_EXTRA3 3
 - (e) SR1_PARAMS 10
 - (f) SR1_POSITION 1
 - (g) SR1_RAW_CTRL 0
 - (h) SR1_RAW_SENS 10
 - (i) SR1_RC_CHAN 0
5. Find the command node **SERIAL1** and set:
 - (a) SERIAL1_BAUD 57
 - (b) SERIAL1_PROTOCOL 1
 6. Click **Write Params**
 7. **Power OFF** and **Power ON** your Pixhawk in order to the changes take effect.

4.2 Using SERIAL 4/5 Port

1. Find the command node **SERIAL4** and set:
 - (a) SERIAL4_BAUD 57
 - (b) SERIAL4_PROTOCOL 1
2. Click **Write Params**
3. **Power OFF** and **Power ON** your Pixhawk in order to the changes take effect.

5 Loading the LUA telemetry scripts to the Taranis X9D transmitter

1. Ensure that Taranis transmitter is turned off;
2. Enter the bootloader mode by sliding both horizontal trims, each under the main sticks, to the center of the remote and then turn on the radio. The Taranis bootloader screen should show;

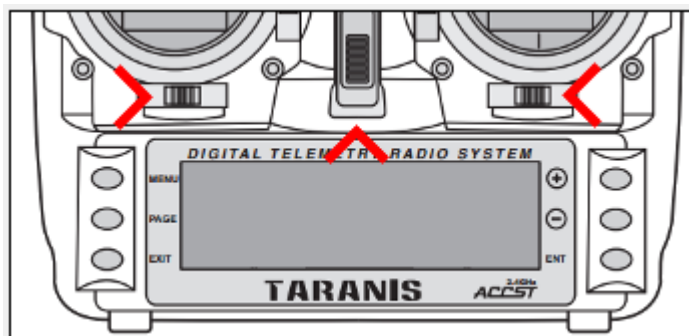


Figure 1: Bootloader mode

3. Connect the Taranis to the computer through the mini-USB located at the back of the transmitter. The text "USB Connected" will be displayed;
4. 2 Removable Drives should appear. One named **Taranis** and the other has a non-descriptive name. The one with the non-descriptive name should be opened;

5. Go to **SCRIPTS** directory. There should have another directory called **TELEMETRY**. If don't, create it and go inside it;
6. Unpack the downloaded *.zip* file mentioned in the **Assumptions** section. Go to the directory **Lua** > > **2.1.X** and copy both **telem1.lua** and **telem2.lua** to the **TELEMETRY** folder mentioned before.
7. Safely remove both drives, turn off the Radio and remove the USB cable.
8. Turn on the radio normally.
9. From the initial screen screen press **MENU** and then long press **PAGE** button to go to the **TELEMETRY** page.
10. Scroll down the **TELEMETRY** page until find the option '**Discover new sensors**'. Toggling this option allows the radio to enable or disable new sensor discovery. For the Airborne Projects' Mavlink Converter 17 sensors should be discovered;
11. Stop the discovery when 17 sensors are discovered;
12. Scroll further down until you find the text **Screen 1**. On **Screen 1** press **ENT** and with the "+" or "-" keys navigate the menu until find the text "Script". When you find it press **ENT**.
13. In the same row, in the next column press **ENT**. A menu with the script names you have placed in the **TELEMETRY** folder will appear. For the **Screen 1** you should select **telem1**;
14. Repeat the steps **12** and **13** on **Screen 2** but now select **telem2**;
15. Press **EXIT** until the initial screen shows up again.
16. Long pressing **PAGE** will lead the telemetry screen with the most information. Pressing again will lead to the 2nd telemetry screen which contains the GPS coordinates and a log of the MAVLink text messages issued by the Autopilot.

6 Troubleshooting

If for some unknown reason the data being displayed on the telemetry screen become inconsistent, please repeat steps **9** and **10** of the previous section.

- All sensors should be deleted before trying to discover new ones (one way to know to be sure there are no "dangling" inactive sensors is that no more than 17 sensors should be displayed).
- This behavior may happen when the Ardupilot is disarmed or when the converter is turned off manually and then reconnected;

No further manual sensor configuration should be necessary as the LUA scripts were designed to work-around the default values that OpenTX presents.

If you have a configuration problem that you cannot solve or you have found a possible bug, please let us know by contacting us to info@airborneprojects.com.

7 Further Information

For more information about the loaded firmware source files and scripts, please visit: <https://github.com/ptsneves/FrSkyTelemetry>.

For customer support, contact us at info@airborneprojects.com.

Appendix A How to update the Converter's firmware

This section only applies for the Pixhawk/Navio versions. All Pixhawk/Navio converters bought before 21th October, 2015 should also be updated.

The following procedure should be performed:

1. Download XLoader from <http://russemotto.com/xloader/XLoader.zip> and unpack it;
2. **VERY IMPORTANT:** Disconnect the converter from any device including from the SmartPort plug and from the DF13 autopilot telemetry plug;
3. Connect the mini-USB cable to the Converter's USB port and wait for it to be recognized by your operating system. It should be recognized as an Arduino;
4. Launch XLoader and configure it according to Figure 2. Taking into account the path where the new firmware (HEX) file is;

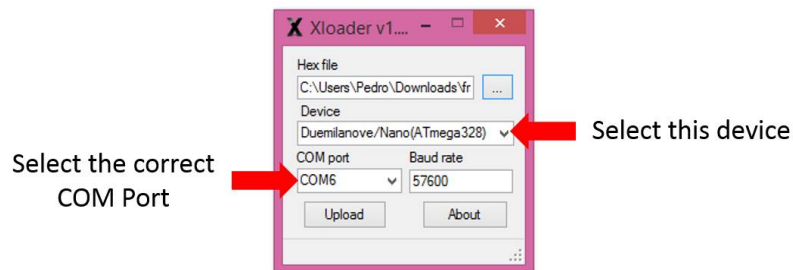


Figure 2: XLoader configuration

5. Hit upload and wait until it is done;
6. Connect the Converter back to your X8R SmartPort and Telemetry Port.

Note: A good way to check if the update was successful is to see if the LED light blinks periodically. If it does a valid firmware is inside. If not, no hard trouble can happen and just try to correctly repeat the above steps.

Appendix B GUI Description

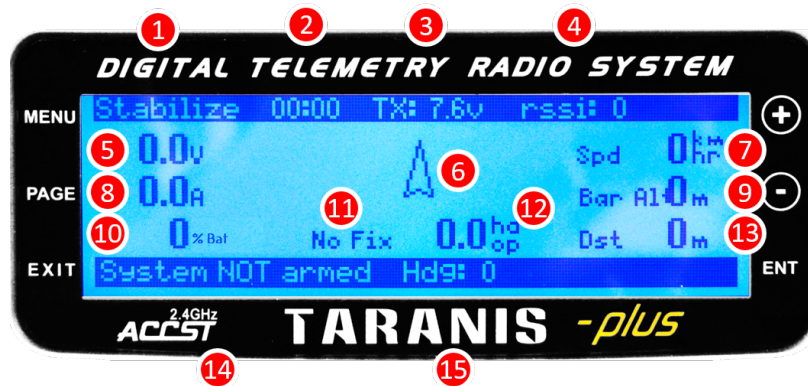


Figure 3: LCD description

1. Flight Mode
2. Taranis user Timer (optional)
3. Taranis Battery Voltage
4. RSSI / radio signal quality
5. UAV Battery Voltage (should be calibrated through Mission Planner for this to be meaningful)
6. Relative Heading Arrow (when system is not armed points measured heading i.e. relative to North; when armed, displays the heading relative to the arming orientation)
7. GPS ground speed (only shows when GPS signal is acquired or enabled)
8. Measured battery current draw (like voltage, must be calibrated through Mission Planner);
9. Barometric Altitude
10. Percentage (%) of battery left (be sure you have both voltage and current calibrated before relying on this)
11. GPS status and/or number of satellites available
12. HDOP (horizontal dilution of precision is a measure of the geometric quality of a GPS satellite configuration in the sky, the lower the better)
13. Distance between actual position and the first GPS position received (GPS locking position)
14. Arming status issued by APM
15. Numeric representation of the arrow angle (6)

Appendix C Product Description

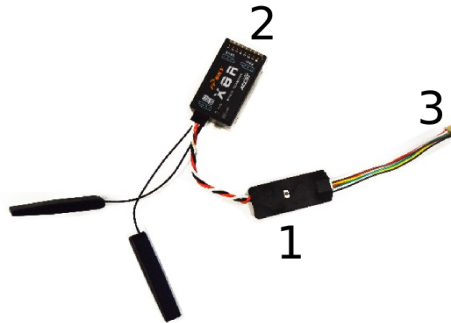


Figure 4: Assembled Converter

1. MavLink to FrSky SmartPort Converter
2. FrSky X8R receiver with Smart Port
3. Telemetry (UART) DF13 cable

Appendix D Specifications

- Translation of Mavlink messages to FrSky Smart Port format;
- Clean and user-friendly telemetry script that detects APM arming to provide relative heading regarding arming orientation, direct from the APM IMU;
- Cheaper alternative to Teensy;
- Lower Power Consumption (20mA);
- Powered directly by the 5V provided by telemetry port;
- Upgradable Firmware (only available on the Pixhawk/Navio version)
- Heat-shrunked wrapping to afford debris protection without compromising weight.