

expresslrs-mavlink.md

```
---
template: main.html
description: ExpressLRS Bi-directional MAVLink support
---
```

!Software Banner

(<https://raw.githubusercontent.com/ExpressLRS/ExpressLRS-Hardware/master/img/software.png>)

```
!!! note "NOTE"
    Enabling MAVLink forces the use of Hybrid or 16ch/2 switch mode. Wide switch mode is not supported.
    Enabling MAVLink forces a telemetry ratio of 1:2
```

Overview

ExpressLRS provides full bi-directional MAVLink (<https://mavlink.io/en/>) support, enabling both native MAVLink telemetry downlink **and** RC control uplink over a single radio link. This integration allows you to use ExpressLRS hardware as a complete replacement for traditional telemetry radios while maintaining RC control functionality.

```
<figure markdown>
!Mavlink-RC Summary (../assets/images/mavlink_rc_summary.png)
</figure>
```

Key Benefits

- **Single radio link and UART** for both RC control and telemetry (no separate telemetry radio needed), and
- **Better range** than traditional telemetry radios thanks to LoRa modes
- **Stubborn sender** ELRS uses its stubborn telemetry sender system for MAVLink messages - this system retr
- **Full parameter editing** and mission planning support
- **Compatible with all major GCS software** (Mission Planner, QGroundControl, etc.)

Hardware Requirements

To start using MAVLink, you just need one ESP based ELRS transmitter, and one ESP based ELRS receiver. The majority of recent ELRS hardware is ESP based, and will be compatible with MAVLink. An easy way to check is, if the RX/TX have WiFi, then it is ESP based. STM32 based hardware cannot support running MAVLink mode.

```
!!! note "NOTE: Performance"
    **Recommendation:** Use 2.4GHz or dual-band-capable (LR1121) hardware for MAVLink whenever possible. If

!!! note "NOTE: Internal TX modules"
    If you are using an internal TX module (e.g. a TX16S with internal ELRS), you will be required to use th
```

Flashing ELRS for MAVLink

Ensure both your transmitter module, the embedded TX-Backpack (if using WiFi forwarding), and your receiver are up to date with the latest release versions:

- For the main ELRS firmware, follow the Firmware Update Guide (<https://www.expresslrs.org/quick-start/getting-started>)
 - For the TX-Backpack, follow the Transmitter Backpack Firmware Guide (<https://www.expresslrs.org/hardware/backpack>)
- The minimum versions to use this feature are:
- Transmitter / receiver firmware: `3.5.0`
 - TX Backpack firmware: `1.5.0`

Configuring ELRS TX & RX for MAVLink

1. Turn on both the RX and TX, and ensure they are bound, connected, and communicating. In the ELRS Lua script look for a C in the top-right corner of the screen which indicates that the TX module and receiver are connected and communicating.

1. Set the protocol on the RX UART to MAVLink.
 - For ELRS v3, with the TX and RX still connected, in the ELRS Lua script, select `Other Devices`, select
 - For ELRS v4, this step can be skipped. RX protocol is now configured automatically when changing the T

1. Turn off your receiver, link mode cannot be changed when connected. In the main menu select the `Link Mod

1. Configure your usual power, packet rate, etc. The telemetry ratio will be preset to 1:2 (and cannot be changed while in MAVLink mode).

1. Power on your receiver and verify it is connected and communicating with the transmitter. As above look for the C in the top-right.

1. Wire the RX to a free UART on your flight controller that is suitable for TLM + RC. A DMA-capable UART is recommended.

Flight Controller Setup

```
=== "ArduPilot"
    For the below steps, when a UART connection is mentioned, it will be written as `SERIALx`. Replace `x` w
```

1. Configure ``SERIALx_PROTOCOL=2``, ``SERIALx_BAUD=460``, and ``RSSI_TYPE=5``
1. Configure all ``SRx_`` parameters to be ``1``, except for ``SRx_ADSB``, ``SRx_PARAMS``, ``SRx_RAW_CTRL`` that s
1. Save parameters and reboot the flight controller

=== "PX4"

For the below steps, when a UART connection is mentioned, it will be written as ``SER_TELx``. Replace ``x`` with

1. Configure `SER_TELx_BAUD` to ``460800 8N1``
1. Configure MAVLink with `MAV_0_CONFIG` to ``TELEM2``
1. Configure MAVLink sending rate with `MAV_0_RATE` to ``9600 B/s``
1. Save parameters and reboot the flight controller

=== "INAV"

Released versions of INAV (from version 8 onwards) can use MAVLink for flight monitoring. However, given

For the below steps, when a UART connection is mentioned, it will be written as ``UARTx``. Replace ``x`` with

1. In the Ports tab, set the UARTx ``Telemetry Protocol`` to ``MAVLink``, and the baud rate to ``460800``. Mak
1. In the Receiver tab, set the Receiver type to ``SERIAL`` and the ``Serial Receiver Provider`` to ``MAVLink``
1. In the CLI, enter ``set mavlink_radio_type=ELRS`` to get the correct RSSI and LQ values on INAV's 0
1. Save and reboot the flight controller.

=== "Betaflight"

Betaflight 2025.12.0-beta and later includes improved MAVLink support with software flow control (via th

For the below steps, when a UART connection is mentioned, it will be written as ``UARTx``. Replace ``x`` with

1. In the Ports tab, on the UARTx row, make sure ``RX Serial`` is enabled, set the ``Telemetry Output`` to ```
1. In the Receiver tab, set the Receiver type to ``Serial (via UART)`` and the ``Serial Receiver Provider``
1. Save and reboot the flight controller.

Handset Setup

Once your flight controller and ExpressLRS hardware are configured for MAVLink, you should have RC control of your craft from your handset. Setup your RC and switch channels as normal in the EdgeTX model screens.

Handset Telemetry

ELRS converts the MAVLink telemetry into standard CRSF format which is understood by EdgeTX. On the Telemetry page in EdgeTX, click "Discover new" to see GPS, battery stats, flight mode, etc.

Yaapu Lua script does not work with ELRS MAVLink in ``v3.x.x`` but is supported in ``v4.x.x``.

Connecting to the Ground Control Station

You are now ready to connect your Ground Control Station (GCS) software to the TX module. ExpressLRS supports two connection methods: ****Serial (USB)**** and ****WiFi (UDP)****. Choose the method that best suits your setup.

Connection Methods Overview

Method	Use Case	Pros	Cons
----- ----- ----- -----			
Serial (USB)	Direct wired connection	Reliable and easy setup, no WiFi needed, works anywhere	Req
WiFi (UDP)	Wireless connection via TX Backpack	Cable-free, mobile GCS operation, works with intern	

Serial (USB) Connection

Serial connection provides a direct, reliable link between your GCS and the TX module via USB cable. This is the recommended method for first-time setup, assuming your TX module supports it (not suitable for internal modules).

Mission Planner Setup (Serial)

1. ****Connect Hardware****:
 - Connect your TX module to your computer via USB cable
 - Ensure the TX module is powered on and bound to your receiver
 - Power on your aircraft so that the TX and RX are linked
1. ****Configure Connection****:
 - Launch Mission Planner
 - In the top-right connection panel, click the dropdown menu next to "Connect"
 - Select the COM port corresponding to your TX module
1. ****Set Baud Rate****:
 - Select ``460800`` from the baud rate dropdown
 - This is the fixed baud rate for ExpressLRS MAVLink
1. ****Connect****:
 - Click the ****Connect**** button
 - Mission Planner should connect and begin receiving telemetry
 - The HUD should display flight data, GPS position, battery voltage, etc.

QGroundControl Setup (Serial)

If you are on Windows, QGroundControl requires a recent daily build to support the DTR USB settings needed for ExpressLRS hardware.

1. ****Download and Install****:

- Download the latest daily build from QGroundControl Daily Builds (<https://docs.qgroundcontrol.com/mast>)
 - Install and launch QGroundControl
1. ****Disable Auto-Connect****:
 - Click the ****Q icon**** (Application Settings) in the top-left corner
 - Select ****Comm Links**** from the sidebar
 - ****Turn off all the auto-connect toggles**** at the top of the page
 - This prevents QGC from connecting with the wrong USB settings
 1. ****Create New Connection****:
 - Click the ****Add**** button (bottom of the screen) to create a new comm link
 - Give it a descriptive name like "ExpressLRS TX Module"
 1. ****Configure Serial Link****:
 - Set ****Type**** to `Serial`
 - Connect your TX module via USB
 - Select your ****COM/Serial port**** from the dropdown:
 - ****Windows****: COM port number (e.g., COM3, COM5)
 - ****Linux****: `/dev/ttyUSB0` or `/dev/ttyACM0`
 - ****macOS****: `/dev/cu.usbserial-\*`
 - Set ****Baud Rate**** to `460800`
 1. ****Configure DTR (Critical Step)****:
 - Click ****Advanced Settings**** to expand additional options
 - Configure based on your operating system:
 - ****Windows****: ☒ Check the `Force DTR Low` checkbox
 - ****Linux****: ☒ Leave the `Force DTR Low` checkbox ****unchecked****
 - ****macOS****: ☒ Leave the `Force DTR Low` checkbox ****unchecked****

!!! danger "Windows Users: DTR MUST Be Low"
 On Windows, you ****must**** enable "Force DTR Low" or the connection will fail and your TX module will
 1. ****Save and Connect****:
 - Click ****OK**** to save the comm link
 - Select your new comm link from the dropdown
 - Click ****Connect****
 - QGroundControl should connect and display vehicle telemetry

WiFi Connectivity

For TX modules that have an onboard Backpack, the MAVLink data can be sent / received via WiFi to a PC or tablet etc. that is running the GCS software. This provides for a cable-free solution on your handset, as it removes the need for a USB cable between the TX and the GCS device.

A Note on WiFi Network Modes

The Backpack supports two WiFi modes:

- ****Access Point (AP) Mode****: Creates its own WiFi network with SSID `ExpressLRS TX Backpack XXXXXX`
 - Best for field use where no existing WiFi network is available
 - Limited range (typically 5-10m depending on environment)
 - Direct WiFi connection between GCS device and TX Backpack
- ****Home WiFi (STA) Mode****: Connects to your home WiFi network
 - Best for bench and indoor use
 - Uses you existing home WiFi infrastructure for extended range
 - GCS and Backpack must be on the same network

Setup WiFi Forwarding for MAVLink

1. Ensure the TX Backpack is running version `1.5.0` or later

1. In the ELRS Lua script, select `Backpack`, scroll to the `Telemetry` option, and change the value to `WiFi`

!!! danger "Attention!"

DO NOT manually put the TX Backpack into WiFi mode via `WiFi Connectivity` -> Enable Backpack WiFi

When you set `Telemetry` to `WiFi` mode in the ELRS Lua script, the Backpack automatically:

- Starts WiFi mode
- Configures the WiFi service for MAVLink forwarding
- Sets up the UDP ports correctly

Manual WiFi enable puts the Backpack in firmware update mode, not MAVLink mode!

1. If you are NOT using Home WiFi on the Backpack, a new WiFi access point will be available to connect to:
 - SSID format: `ExpressLRS TX Backpack XXXXXX` (where `XXXXXX` is part of your module's UID)
 - WiFi password: `expresslrs`
 - The Backpack will use IP `10.0.0.1` in AP mode

Alternatively, if you ARE using Home WiFi on the Backpack, the Backpack will automatically start WiFi and

1. Connect your PC or tablet to the Backpack's WiFi network (or the same home network)

1. Power on the craft so that the TX and RX have an active link

1. Make sure no firewall is blocking UDP traffic (e.g. Linux: disable `ufw`, Android: select "use network as is")

1. Configure your GCS software:

Mission Planner Setup (UDP)

1. ****Connect to Backpack WiFi****:
 - Connect your computer to the TX Backpack's WiFi network
 - SSID: `ExpressLRS TX Backpack XXXXXX` (where XXXXXX is part of your UID)
 - Password: `expresslrs`
 - Or connect to your home network if using STA mode
1. ****Configure UDP Connection****:
 - Launch Mission Planner
 - In the top-right connection panel, click the dropdown and select ****UDP****
 - Click the ****Connect**** button
 - A dialog box will appear asking for the port number
1. ****Set Port****:
 - Enter `14550` (the default MAVLink UDP port)
 - Click ****OK****

QGroundControl Setup (UDP)

1. ****Connect to Backpack WiFi****:
 - Connect your device to the TX Backpack's WiFi network
 - SSID: `ExpressLRS TX Backpack XXXXXX`
 - Password: `expresslrs`
 - Or connect to your home network if using STA mode
1. ****Enable Auto-Connect****:
 - Open QGroundControl
 - Click the ****Q icon**** (Application Settings)
 - Select ****Comm Links****
 - Turn on the auto-connect toggle for UDP (it should be on by default unless you have disabled it)
 - QGroundControl should automatically connect and receive telemetry as soon as your Backpack starts sending

Best Practices and Tips

Choosing Packet Rates

For MAVLink, higher packet rates provide better telemetry data speeds.

Suggested settings:

- ****2.4GHz****: Best speeds will be achieved with F1000 packet rate. We recommended that you use F1000 or 333Hz
- ****Dual Band****: Best speeds will be achieved with K1000 packet rate. We recommended that you use K1000 for
- ****Single band 900MHz****: Legacy 900MHz packet rates are NOT recommended. If you insist on using these rates

Throughput per RF Mode

The tables below give the estimated MAVLink data capacity for every packet rate, using the packet-rate names as they appear in the ELRS Lua script. Higher rates and the ***Full*** modes (which use the larger 13-byte over-the-air packet) carry more MAVLink data.

The ****Uplink (TX→RX)**** column assumes RC channels are also being sent (the normal case), where MAVLink shares the uplink with RC packets. The ****Uplink (no RC)**** column is the higher rate available when running standalone with no handset/RC, so the full uplink is available to MAVLink.

2.4GHz

Packet Rate	RX Sensitivity	OTA Packet	Downlink (RX→TX)	Downlink (Gemini)[^gemini]	Uplink (TX→RX)
50Hz	-115 dBm	8 B	~110 B/s	~225 B/s	~60 B/s
100Hz Full	-112 dBm	13 B	~435 B/s	~870 B/s	~220 B/s
150Hz	-112 dBm	8 B	~350 B/s	~700 B/s	~180 B/s
250Hz	-108 dBm	8 B	~590 B/s	~1175 B/s	~300 B/s
333Hz Full	-105 dBm	13 B	~1470 B/s	~2935 B/s	~735 B/s
500Hz	-105 dBm	8 B	~1185 B/s	~2370 B/s	~595 B/s
F500	-104 dBm	8 B	~1185 B/s	~2370 B/s	~595 B/s
F1000	-104 dBm	8 B	~2375 B/s	~4755 B/s	~1190 B/s
D250	-104 dBm	8 B	~595 B/s	~1190 B/s	~300 B/s
D500	-104 dBm	8 B	~1190 B/s	~2375 B/s	~595 B/s
K1000	-103 dBm	8 B	~2375 B/s	~4755 B/s	~1190 B/s
DK500	-103 dBm	8 B	~1190 B/s	~2375 B/s	~595 B/s
DK250	-103 dBm	8 B	~595 B/s	~1190 B/s	~300 B/s

900MHz

Packet Rate	RX Sensitivity	OTA Packet	Downlink (RX→TX)	Downlink (Gemini)[^gemini]	Uplink (TX→RX)
25Hz	-123 dBm	8 B	~50 B/s	~105 B/s	~30 B/s
50Hz	-120 dBm	8 B	~110 B/s	~225 B/s	~60 B/s
D50Hz	-112 dBm	8 B	~115 B/s	~235 B/s	~60 B/s
100Hz	-117 dBm	8 B	~230 B/s	~460 B/s	~120 B/s
100Hz Full	-112 dBm	13 B	~435 B/s	~870 B/s	~220 B/s
200Hz	-112 dBm	8 B	~470 B/s	~940 B/s	~240 B/s
200Hz Full	-111 dBm	13 B	~880 B/s	~1760 B/s	~445 B/s

250Hz	-111 dBm	8 B	~590 B/s	~1175 B/s	~300 B/s
K1000 Full	-101 dBm	13 B	~4420 B/s	~8845 B/s	~2215 B/s

Dual Band (LR1121)

Dual band always runs Gemini (one radio per band), so the downlink figures below already include the doubling.

Packet Rate	RX Sensitivity	OTA Packet	Downlink (RX→TX)	Uplink (TX→RX)	Uplink (no RC)	Chip set
----- :----- :----- :----- :----- :----- :-----						
100Hz Full	-112 dBm	13 B	~870 B/s	~220 B/s	~445 B/s	LR1121 (d
150Hz	-112 dBm	8 B	~700 B/s	~180 B/s	~360 B/s	LR1121 (d

??? info "How these numbers are calculated"

MAVLink mode forces a 1:2 telemetry ratio, so half of all packet slots are the downlink (telemetry) and

Per mode, with effective packet rate $F = 1e6 / (\text{interval_us} \times \text{numOfSends})$, OTA payload $P = 5$ bytes (

...

U(P) = $62 / \text{ceil}(64 / P)$ # 2-byte message header + partial-chunk framing
P_dl = $((P - 4) + B \cdot P) / (B + 1)$ # one LinkStats frame (4 B) per B data frames

Downlink [B/s] = $(F / 2) \cdot U(P) \cdot (P_dl / P)$
Downlink (Gemini) = $2 \cdot \text{Downlink}$ # two radios send different halves
Uplink [B/s] = $(F / 4) \cdot U(P)$ # shared with RC channel packets
Uplink (no RC) [B/s] = $(F / 2) \cdot U(P)$ # standalone, full uplink available

The acknowledgement is the single `stubbornAck` bit carried in each packet's header (no extra bytes).

[^gemini]: Downlink rate on dual-radio hardware running **Gemini** mode (needs two same-band radios and the Gemini antenna mode). The two radios send different halves of the telemetry stream instead of redundant copies, doubling the downlink. Uplink is unaffected. Dual-band modes always run Gemini, so their downlink figures already include this.

Advanced Features

MAVLink System IDs (Optional)

MAVLink uses System IDs (SysID) to identify different devices on the network. ExpressLRS allows you to configure these IDs to match your setup:

- **Source SysID**: The System ID that the ExpressLRS receiver uses when acting as a GCS (Ground Control Station)
- **Target SysID**: The System ID of your flight controller/vehicle. Default is `1`, which is the standard v

!!! info "When to Change System IDs"

In most cases, the default values (Source: 255, Target: 1) work perfectly. You may need to change the

- You have multiple vehicles on the same MAVLink network
- Your flight controller uses a non-standard System ID
- You're integrating with other MAVLink devices (companion computers, gimbals, etc.)

To configure System IDs:

1. Set the receiver to MAVLink mode (as described above)
1. Power on the TX and RX so that they are linked (solid LED on the RX)
1. Open the ELRS Lua script, navigate to `Other Devices` and select your receiver
1. Set `Target SysID` to match your vehicle's System ID (typically `1`)
1. Set `Source SysID` to your preferred GCS System ID (typically `255`)
1. Save and reboot

The receiver will now use these IDs for all MAVLink communication, ensuring proper message routing in complex MAVLink networks.

Standalone Telemetry Radio Mode

ExpressLRS TX modules can be used as standalone telemetry radios, similar to traditional SiK radios, **without** requiring a connection to a handset. This mode is useful for:

- Fixed-wing long range flights where you only need telemetry (separate RC control)
- Bench testing and configuration
- Autonomous missions where RC control is not needed

How to Use Standalone Mode

1. Before using the TX module standalone, you will need to setup the packet rate, RF power, etc. using a handset and the ELRS Lua script. You can also set the `Link Mode` option to `MAVLink` if you want, although this is not strictly necessary as the TX will detect MAVLink and change this setting automatically.

!!! danger "IMPORTANT!"

Make sure you are using Model ID 0 in EdgeTX when you do this setup. ELRS uses the settings that are

1. Connect the TX module to your computer via USB (no handset needed)

1. Power on your aircraft with the receiver

1. Configure your GCS software (Mission Planner, QGroundControl, etc.) to use the TX module's COM port at `460800` baud

1. Click connect in your GCS. The TX module will **automatically detect** MAVLink packets from your GCS software and start the link

- The module monitors the USB connection for incoming MAVLink traffic
- When MAVLink packets are detected, it automatically starts the radio link

1. The module now functions as a bi-directional telemetry radio, forwarding MAVLink between your GCS and the aircraft

TX Backpack MAVLink Statistics

When using WiFi forwarding through the TX Backpack, the Backpack tracks additional statistics and displays them on the Backpack WebUI, on the MAVLink tab (go to `http://10.0.0.1`, or `http://elrs\_txbp.local/` if using home WiFi).

- **Downlink packets**: Count of MAVLink messages from aircraft to GCS
- **Uplink packets**: Count of MAVLink messages from GCS to aircraft
- **Dropped packets**: Number of messages with sequence number gaps (indicates RF link issues)
- **Buffer overflows**: Number of times the 16-message buffer filled up (indicates too much telemetry for av

These statistics help diagnose performance issues.