

Восстановление обычного ExpressLRS

Методички

MILELRS → ExpressLRS

Приёмник BETAFPV SuperD 2.4GHz RX, ESP32, плата V1.1 с фотографии. Полная установка через USB-UART, резервная копия и проверка связи. Приложение: внешний BETAFPV 900MHz Micro TX V2 из учебного примера.

Windows 10/11, Chrome или Edge. Подготовлено 12.09.2026.

Что должно получиться

RX запускает обычный ExpressLRS, связывается с совместимым TX, после повторного включения привязка сохраняется, а в Betaflight видны движения стиков и переключателей.

TX - радиопередатчик в пульте или внешний модуль. **RX** - радиоприёмник на дроне. **FC** - полётный контроллер, который получает команды от RX. У каждого своя прошивка. Для замены MILELRS в SuperD не требуется перепрошивать FC или EdgeTX в пульте.

Target - описание конкретной платы для сборки прошивки. Для показанного RX выбирать **BETAFPV SuperD 2.4GHz RX**. В каталоге это `betafpv.rx_2400.superd`; ESP32 и поддержка UART подтверждены [официальным каталогом](#).

В комплекте два самостоятельных маршрута:

- **RX. SuperD напрямую через USB-UART.** Основная инструкция для напарника: сохранение FC и памяти RX, запись, Bind и проверка.
- **TX. BETAFPV Micro TX V2 через USB.** Способ для рассматриваемого внешнего модуля 900 MHz; это отдельный пример. Его target не подходит TX 2.4 GHz и он не свяжется с SuperD 2.4 GHz.

Согласовать версию с пультом

1. Включить пульт с подключённой антенной TX.
2. Открыть меню ExpressLRS и прочитать строку версии внизу; у модуля с экраном версия может быть показана и там.
3. Записать название TX, частоту, версию и регион FCC/LBT, если он указан.

Что стоит на TX	Какую прошивку выбрать для SuperD
Обычный ExpressLRS 3.x, TX оставляем	ExpressLRS 3.6.4; та же основная версия 3.x
Обычный ExpressLRS 4.x	ExpressLRS 4.1.0
MILELRS или версия не установлена	Сначала определить TX. Для перехода обеих сторон на 4.1.0 нужен отдельный target этого TX.

SuperD на фото - **2.4 GHz**: TX тоже должен поддерживать этот диапазон. Обычные ELRS 3.x и 4.x несовместимы. Номер MILELRS 3.76 не доказывает совместимость с ELRS 3.x. [Релизы ExpressLRS](#).

[[pagebreak]]

Почему выбран прямой UART

USB-UART соединяет USB компьютера с сигнальными выводами RX. В режиме **BOOT** запускается встроенный загрузчик ESP32, принимающий прошивку по проводу. Доступ к Wi-Fi и ключи прежней MILELRS для этого не нужны; прежние ключи при этом не восстанавливаются.

Betaflight Passthrough тоже поддерживается SuperD: FC передаёт данные между USB и RX. Но текущий Web Flasher для ESP32 этим способом записывает только приложение и отключает полное стирание. Несовместимая разметка памяти или отказ MILELRS принять команду загрузчика могут помешать такой попытке. Это возможные причины, а не установленный диагноз приёмника напарника. [Код маршрута](#).

Прямой Serial UART устанавливает приложение, загрузчик и таблицу разделов. Адреса выбирает программа. Отдельный BIN из Local Download для этого ESP32 обычно содержит только приложение: его нельзя считать полной копией памяти или записывать как полный образ с адреса 0. [Формирование образов](#), [Local Download](#).

Резервирование и привязка

Full chip erase стирает память радиоустройства, включая старую прошивку, Wi-Fi и привязку. Сначала сохраняем полный файл через **read-flash**. Копия diff all хранит настройки FC, копия SD - настройки пульта; они не заменяют копию RX/TX.

Binding Phrase создаёт идентификатор пары, это не пароль шифрования. Для новой пары придумать свою фразу от 8 латинских букв/цифр и одинаково записать на TX/RX. Если рабочий TX сохраняется, использовать его фразу; если она неизвестна, RX можно связать вручную. [Привязка ExpressLRS](#).

В Web Flasher для SuperD поле **Region** предлагает **FCC** и **LBT**. Они соответствуют ISM2G4 и EU_CE_2400 в других инструментах ELRS. Выбирать вариант, совместимый с TX и применимым местным режимом эксплуатации. Если регион неизвестен, прочитать конфигурацию TX до записи. FCC915 относится к рассматриваемому примеру пары 900 MHz. [Поле Region](#), [региональные опции](#).

Границы проверки

Технический пример, проверенный 12.09.2026: BETAFPV 900MHz Micro TX V2 и HappyModel ES900RX, переход MILELRS v3.76 → ExpressLRS 4.1.0. Запись и реакция каналов после Bind подтверждены в этой практике; автоматическая привязка без Bind не подтверждена. Это не описание текущего оборудования собеседника.

В той практике запись выполнялась через esptool и инструмент ExpressLRS; браузер использовался для подготовки образов. Инструкция сверена с официальными документами, интерфейсом и исходниками. **Этот маршрут на SuperD не испытан; подтверждена сверка по документации и исходникам.** Синтаксис команд проверен по esptool 5.4.0. При подготовке документа оборудование не перепрошивалось.

RX. SuperD через USB-UART

Для **BETAFPV SuperD 2.4GHz RX на ESP32**, включая показанную плату **V1.1**. Это UART-приёмник; он прошивается отдельно от Betaflight. Цель - полная замена MILELRS на обычный ELRS с сохранением исходной памяти в файл.

Подключение FC: UART определяется по фактической пайке и настройкам платы. UART2/T2/R2 далее — учебный пример, не готовая конфигурация чужого борта. Для UART-приёмника ELRS используются Serial RX и CRSF; значения inversion и half-duplex сверяются отдельно.

1. Что приготовить

- Windows 10/11, Chrome/Edge и интернет.
- USB-UART с **логикой TX/RX 3.3V** и отдельным выходом питания **5V**: адаптер CP210x/FT232 с документированными уровнями либо [BETAFPV ExpressLRS Recovery Dongle](#).
- USB-кабель данных, 4 коротких провода, паяльник при паяном подключении, мультиметр и доступ к BOOT.
- Target **BETAFPV SuperD 2.4GHz RX**. Совместимый TX 2.4 GHz; RX 3.6.4 для TX 3.x либо RX 4.1.0 для TX 4.x.

Питание SuperD - **5V на «+»**. Аккумулятор дрона напрямую сюда не подключать. Питание 5V и уровень сигналов 3.3V - разные параметры: переключатель адаптера «5V» иногда меняет и уровень TX. Проверить паспорт адаптера; при сомнении измерить его TX относительно GND в покое. [Паспорт SuperD](#).

2. Сохранить исходное подключение и FC

1. Отключить аккумулятор дрона.
2. Отключить USB дрона.
3. Снять пропеллеры.
4. Сфотографировать все 4 соединения RX с FC так, чтобы восстановить каждое.
5. Сверить пайку с настройкой из дампа: например, для **UART2** нужны сигнальные площадки **T2/R2** FC.
6. Подключить USB к разъёму FC.
7. В [Betaflight](#) или совместимом установленном Configurator подключиться к появившемуся порту кнопкой **Connect**.
8. Открыть **CLI**.
9. Выполнить `diff all`.
10. Сохранить весь ответ в `fc-before-elrs.txt` в отдельной папке этого дрона.
11. Нажать **Disconnect**.
12. Закрыть Betaflight.
13. Отключить USB FC.

Если FC недоступна, сохранить фото и имеющуюся предыдущую копию настроек. Это не препятствует прямой прошивке снятого RX; проверка неизвестных настроек FC после возврата RX остаётся обязательной.

[[pagebreak]]

3. Подключить SuperD к адаптеру

На схеме **FC Rx / FC Tx** обозначают выводы полётного контроллера. Площадка приёмника напротив **FC Rx - TX самого приёмника**, напротив **FC Tx** - его **RX**. Это важно при переносе на адаптер.

[[image:superd]]

Источник: [ExpressLRS, SuperD 2.4GHz](#). Вид со стороны ESP32; BOOT отмечена сверху справа. Сверить ориентацию платы с рисунком, а не с цветами проводов на другом дроне.

1. Убедиться, что аккумулятор и USB FC отключены.
2. Отсоединить **все 4 провода RX от FC**; при пайке удобно оставить их на RX и освободить концы со стороны FC.
3. Закрепить RX на непроводящей поверхности с доступной BOOT.
4. Подключить GND адаптера к GND RX.
5. Подключить TX адаптера к RX приёмника.
6. Подключить RX адаптера к TX приёмника.
7. Подключить выход 5V адаптера к «+» RX.
8. Проверить отсутствие перемычек припоя между соседними площадками.

[[diagram:uart]]

Обе антенны оставить подключёнными. USB FC, аккумулятор и питание от FC здесь не используются. Замыкать площадки не требуется: у SuperD есть кнопка BOOT.

[[pagebreak]]

4. Войти в BOOT и найти порт

1. Нажать и удерживать **BOOT на RX**.
2. Подключить USB адаптера к компьютеру, продолжая удерживать BOOT.
3. Отпустить BOOT после подачи питания.
4. Открыть **Device Manager → Ports (COM & LPT)** в Windows.
5. Записать COM-порт появившегося адаптера.

RGB-светодиод SuperD в BOOT может не гореть. Режим подтверждается ответом ESP32 в программе. Встречающиеся в инструкциях слова про «solid LED» относятся не ко всем индикаторам. [Особенность SuperD](#).

Если порт неизвестен, сравнить список при отключённом и подключённом адаптере. Жёлтый значок требует драйвера его микросхемы: [CP210x](#) или [FTDI](#). После установки повторить вход в BOOT. DFU - не нужный порт; адаптер должен появиться как COM.

5. Сохранить всю старую прошивку

esptool - официальная программа Espressif для работы с памятью ESP. Сначала выполняется только чтение.

В PDF длинные команды перенесены на несколько строк с обратной кавычкой в конце. Копировать весь блок; это одна команда PowerShell. Команды set и save ниже относятся к CLI Betaflight, а не к PowerShell.

1. Скачать [esptool 5.4.0 для Windows x64](#).
2. В Проводнике выполнить **Extract All** для ZIP.
3. Открыть извлечённую папку esptool-windows-amd64, где находится esptool.exe.
4. В адресной строке этой папки ввести powershell и нажать Enter.

5. Выполнить строку ниже, заменив **COM7** своим портом.

```
$radioPort = 'COM7'
```

6. Выполнить чтение:

```
.\esptool.exe -p $radioPort --before no-reset -a no-reset read-flash 0 ALL rx-original.bin
```

Ожидается определение ESP32, размера flash и окончание чтения без ошибки. ALL означает всю память. В папке появляется rx-original.bin; для штатного ESP32-PICO-D4 ожидается 4 MiB. Другой процессор или неожиданный размер требуют сверки платы до стирания. Чтение при 115200 может идти несколько минут.

7. Проверить копию сравнением с памятью:

```
.\esptool.exe -p $radioPort --before no-reset -a no-reset verify-flash 0 rx-original.bin
```

8. Сохранить rx-original.bin и текст успешного чтения/проверки в папку дрона.

Если verify-flash не смог подключиться повторно, заново войти в BOOT и повторить проверку. При ошибке копия не считается проверенной; до её получения Full chip erase не включать. [Команды esptool](#).

[[pagebreak]]

6. Подготовить ELRS в браузере

1. Открыть [ExpressLRS Web Flasher](#) в Chrome или Edge.
2. Выбрать **Receiver**.
3. В **Firmware Version** выбрать **3.6.4** для TX на ELRS 3.x либо **4.1.0** для TX на ELRS 4.x.
4. В **Hardware Vendor** выбрать **BETAFPV**.
5. В **Radio Frequency** выбрать **2.4GHz Receiver**.
6. В **Hardware Target** выбрать **BETAFPV SuperD 2.4GHz RX**.
7. Нажать **Next**.
8. Нажать **Clear Stored Settings**, чтобы убрать настройки ранее прошивавшегося устройства.
9. В **Bind Phrase** ввести фразу пары; если фраза работающего TX неизвестна, оставить поле пустым для ручного Bind.
10. В **Region** выбрать тот же вариант **FCC** или **LBT**, что используется совместимым TX.
11. Оставить **WiFi SSID** и **WiFi Password** пустыми.
12. В **Flashing Method** выбрать **Serial UART**.

Ожидается SuperD 2.4GHz RX, согласованная версия и Serial UART. Advanced Settings остаются штатными, RX as TX выключен. Wi-Fi-поля задают домашнюю сеть; для записи по проводу они не нужны. [Выбор платы, опции RX](#).

13. Отключить USB адаптера.
14. Снова зажать BOOT на RX.
15. Подключить USB адаптера.
16. Отпустить BOOT.
17. Нажать **Next** в Web Flasher.
18. В запросе браузера выбрать COM-порт **адаптера RX** и подтвердить подключение.

Если запрос отменён, вызвать его кнопкой **Connect**. Ожидается определение ESP32 и доступная Flash. Betaflight и другие программы не должны удерживать этот порт.

7. Выполнить полную запись

1. Убедиться, что проверенная копия rx-original.bin сохранена отдельно.
2. Включить **Full chip erase**.
3. Нажать **Flash**.
4. Дождаться **Done** без ошибки **Flash failed**.

Для этого ESP32 при Serial UART записываются 4 образа; несколько последовательных процентов записи нормальны. Кабель и питание во время записи не отключать. Если нет Full chip erase и доступно только приложение, проверить способ: Serial UART, а не Betaflight Passthrough. [Механизм записи](#).

При **Wrong MCU** остановиться: ESP32-C3, ESP8285 и ESP32 - разные платформы. При **Target mismatch** сохранить лог. Flash Anyway допустим только после проверки платы и аппаратного target, а не как универсальное лечение ошибки.

5. Отключить USB адаптера.
6. Подключить его заново **без нажатой BOOT**.

Ожидается запуск приложения RX. Отсутствие свечения в предшествующем режиме загрузчика было допустимо.

[[pagebreak]]

8. Проверить версию до возврата на дрон

1. Оставить пульт выключенным.
2. Подождать 60 секунд после обычного запуска RX.
3. Подключить ноутбук к Wi-Fi **ExpressLRS RX**, пароль **expresslrs**.
4. Открыть **http://10.0.0.1** в браузере.
5. Проверить **BETAFPV SuperD 2.4GHz RX** и выбранную версию ELRS.
6. Сохранить снимок страницы.
7. Вернуть ноутбук в обычную Wi-Fi-сеть.
8. Отключить USB адаптера.

Это чтение страницы, не повторная прошивка. Быстрое зелёное мигание RGB штатного SuperD соответствует Wi-Fi. Если сети нет, проверить питание и обычный запуск; при настроенной домашней сети RX может подключиться к ней. [Версия RX](#), [инструкция BETAFPV](#).

9. Вернуть RX и проверить FC

1. Отсоединить RX от адаптера.
2. Восстановить соединения с FC по фото: RX.TX → FC.RX, RX.RX → FC.TX, «+» → 5V, GND → GND.
3. Убедиться, что BOOT не зажата креплением или термоусадкой.
4. Подключить USB к FC.
5. Если RX не питается от USB, подключить аккумулятор с охлаждением VTX и установленной антенной VTX.
6. В Betaflight нажать **Connect** для порта FC.
7. В **Ports** проверить **Serial RX на фактически используемом UART**; для примера UART2 пайка соответствует T2/R2.
8. Если Serial RX пришлось исправить, нажать **Save and Reboot**.

9. В **Receiver** проверить **Serial (via UART)** и **CRSF**.

10. Если режим или протокол пришлось исправить, нажать **Save and Reboot**.

11. В **CLI** выполнить `get serialrx`.

Для обычного полного UART требуются `serialrx_inverted = OFF` и `serialrx_halfduplex = OFF`. Если отличаются, выполнить по одной строке:

```
set serialrx_inverted = OFF
set serialrx_halfduplex = OFF
save
```

Если CRSF/OFF/OFF уже установлены, повторная запись этих значений не нужна. В новом формате CLI пример `serial UART2 64 ...` означает Serial RX на UART2; маска 64 означает Serial RX. Фактические точки пайки сверяются с выбранным UART. [Настройка FC](#).

[[pagebreak]]

10. Bind и проверка результата

TX уже должен работать на обычном ELRS той же основной версии, в 2.4 GHz и совместимом регионе. Если он ещё на MILELRS, требуется его отдельная перепрошивка по модели TX.

1. Включить пульт с антенной TX.
2. Открыть **ExpressLRS** в инструментах пульта; на TX16S коротко нажать SYS, выбрать ExpressLRS в Tools и коротко нажать ролик.
3. Подать питание на дрон без пропеллеров.
4. Проверить связь; при одинаковой Binding Phrase она обычно появляется автоматически.

Если связи нет, ручной Bind на ELRS 3.6.4/4.1.0:

5. На **уже включённом RX** удерживать BOOT около **1.5 секунды**, затем отпустить.
6. После двойных вспышек RGB выбрать **Bind** в ExpressLRS пульта и коротко нажать ролик.

Ожидается подключение и **C** в Lua при рабочей телеметрии. Альтернатива кнопке RX при Betaflight 4.5+: **Receiver → Bind Receiver**. Если RX ушёл в Wi-Fi, перезапустить его перед Bind. [Способы привязки](#).

7. Открыть **Receiver** в подключённом Betaflight.
8. Поочерёдно двигать каждый стик.
9. Проверить правильность осей и направления соответствующих полос.
10. Поочерёдно переключить используемые тумблеры и проверить AUX-каналы.
11. Выключить питание дрона.
12. Включить питание дрона без BOOT.
13. Проверить восстановление связи и каналов без нового Bind.
14. При снятых пропеллерах и выключенном ARM выключить TX.
15. Убедиться, что FC показывает потерю приёмника/RXLOSS/FAILSAFE.
16. Включить TX и проверить восстановление приёма команд.

Успех: правильные каналы, сохранённая привязка и распознавание потери связи. Постоянный LED сам по себе этого не доказывает. Здесь проверяется радиоканал на столе; фактическую реакцию дрона на failsafe и ARM сверяют отдельно до полёта.

Если оси перепутаны, исправляют Channel Map или микшеры. Если связь есть, полосы неподвижны - проверяют CRSF и RX.TX → FC.RX. При Model Mismatch проверяют Model Match в Lua; для первичного соединения обычно используется Off. [Проверка перед полётом](#).

11. Если снова не получилось

Где остановилось	Что проверить
COM не появился	Кабель данных/USB-разъём, затем драйвер микросхемы адаптера.
Failed to connect / нет ответа ESP	BOOT удерживалась при подаче питания; RX отключён от FC; 5V на «+»; TX/RX перекрещены; общая GND.
ESP отвечает, запись срывается	Короткие провода, устойчивые 5V, исправный кабель. Сохранить лог; не подбирать другой target.
При попытке через FC программа не находит Serial RX	В новых версиях используется именованный формат serial UART2. Старый инструмент может его не понимать; текущий Web Flasher учитывает этот формат. Прямой UART обходит FC.
После записи красные вспышки	Возможна проблема связи ESP с радиочипами. Проверить target; при верном target нужна диагностика платы.
Работает снятым, не запускается на FC	BOOT/крепление и влияние UART FC: линия может удерживать ESP в загрузчике.
Теряет Bind после перезапуска	Проверить Bind Storage: не должен быть Volatile; заново связать и проверить сохранение.

Передать для разбора полный лог, выбранные версию/target/Region, схему соединения и индикацию после запуска. Повреждённое питание или радиоузел перепрошивка не исправит. [Восстановление ELRS](#), [индикация RGB](#).

12. Вернуть свою исходную копию

Только для отката: rx-original.bin должен быть проверенной полной копией **этого же RX**. Возвращается прежняя MILELRS с настройками; новый ELRS удаляется.

1. Отсоединить RX от FC и подключить к адаптеру по разделу 3.
2. Войти в BOOT по разделу 4.
3. Открыть PowerShell в папке esptool с файлом своей копии.
4. Задать \$radioPort с актуальным COM-портом, как в разделе 5.
5. Выполнить запись полной копии:

```
.\esptool.exe -p $radioPort --before no-reset write-flash --erase-all 0 rx-original.bin
```
6. После успешной записи заново войти в BOOT.
7. Выполнить verify-flash из раздела 5.
8. После успешного сравнения отключить USB и включить RX без BOOT.

Адрес **0** здесь относится к полной копии памяти. Для одиночного приложения ELRS его использовать нельзя. [Операции esptool](#).

TX. BETAFPV Micro TX V2

Отдельный пример для BETAFPV 900MHz Micro TX V2, учебный пример 915 MHz.

Полная установка ExpressLRS 4.1.0/FCC915 через USB самого модуля. Это не инструкция выбора target для неизвестного TX 2.4 GHz.

Нужны Windows x64, кабель USB-C с передачей данных, интернет, microSD-кардридер и [esptool 5.4.0](#). Версия 4.1.0 требует RX с ELRS 4.x того же диапазона. Показанный SuperD 2.4 GHz с этим модулем не работает.

1. Подготовить пульт и сохранить TX

1. Выключить TX16S.
2. Извлечь microSD пульта.
3. Через кардридер скопировать всё содержимое SD в отдельную папку на компьютере.
4. Подключить антенну внешнего TX, если она снята.
5. Отключить внешний аккумулятор модуля, если он использовался.
6. Подключить USB компьютера к **USB-C внешнего модуля**.

[[diagram:tx]]

Ожидается последовательный USB-порт модуля. Пульт остаётся выключенным. Схема относится именно к Micro TX V2; переключатели и разводка старых BETAFPV Micro TX могут отличаться. [Устройство Micro TX V2, руководство BETAFPV](#).

7. В **Device Manager** → **Ports (COM & LPT)** записать COM-порт модуля.
8. Извлечь архив esptool через **Extract All**.
9. Открыть папку esptool-windows-amd64, где находится esptool.exe.
10. В её адресной строке ввести powershell и нажать Enter.
11. Выполнить строку, заменив **COM12** своим портом:

```
$txPort = 'COM12'
```

12. Прочитать всю память TX:

```
.\esptool.exe -p $txPort -b 115200 read-flash 0 ALL tx-original.bin
```

13. Проверить соответствие копии памяти:

```
.\esptool.exe -p $txPort -b 115200 verify-flash 0 tx-original.bin
```

14. Скопировать tx-original.bin и успешный лог в отдельную папку этого модуля.

Ожидается ESP32 и успешное чтение/сравнение. В проверенном учебном примере flash 8 MiB; ALL читает обнаруженный объём. Операция может занять несколько минут. При ошибке копия не считается проверенной; стирание отложить. 115200 выбрано для устойчивого чтения: в проверенном примере чтение при 460800 завершалось ошибкой. [Резервирование esptool](#).

[[pagebreak]]

2. Получить полный комплект 4.1.0

1. Открыть [ExpressLRS Web Flasher](#).
2. Выбрать **Transmitter**.
3. В **Firmware Version** выбрать **4.1.0**.
4. В **Hardware Vendor** выбрать **BETAFPV**.
5. В **Radio Frequency** выбрать **900MHz Transmitter**.

6. В **Hardware Target** выбрать **BETA FPV 900MHz Micro TX V2**.
7. Сохранить Lua по ссылке **Download ELRS Lua Script** на этом экране.
8. Нажать **Next**.
9. Нажать **Clear Stored Settings**.
10. В **Bind Phrase** ввести собственную фразу пары, одинаковую с её RX 900 MHz.
11. В **Regulatory Domain** выбрать **FCC915** для рассматриваемого варианта учебной пары.
12. Оставить **WiFi SSID** и **WiFi Password** пустыми.
13. В **Flashing Method** выбрать **Local Download**.
14. Нажать **Next**.
15. Нажать **Download**.
16. Скопировать полученный BIN в папку с esptool.exe под именем **firmware.bin**.

Это приложение с аппаратным описанием Micro TX V2 и выбранной фразой. Для полной установки нужны ещё 3 файла **из того же релиза и семейства ESP32 900 TX**. В ссылках ниже зафиксирована сборка 4.1.0, использованная в проверенном учебном примере; они не подходят произвольной другой версии или RX.

17. Сохранить в ту же папку [bootloader.bin](#).
18. Сохранить туда [partitions.bin](#).
19. Сохранить туда [boot_app0.bin](#).
20. Проверить, что все 4 BIN лежат рядом с esptool.exe, без добавленного .txt и суффиксов вроде (1).

Если браузер пытается открыть двоичные данные, сохранить ссылку через **Save link as**. Размеры трёх служебных файлов: bootloader 17536 байт, partitions 3072 байта, boot_app0 8192 байта. Это проверка комплекта, не замена проверки записи. [Опции TX](#), [структура образов ESP32](#).

3. Записать 4 образа

1. Убедиться, что копия tx-original.bin проверена и сохранена отдельно.
2. Закрыть программы, использующие COM-порт TX.
3. В ранее открытом PowerShell выполнить **весь блок** ниже:

```
.\esptool.exe -p $txPort -b 230400 write-flash --erase-all `
  0x1000 bootloader.bin `
  0x8000 partitions.bin `
  0xE000 boot_app0.bin `
  0x10000 firmware.bin
```

Обратная кавычка в конце строк - продолжение команды PowerShell; не заменять её апострофом. Полное стирание удаляет старую MILELRS и настройки TX. Именно эти адреса относятся к классическому ESP32 в данном модуле; на ESP32-C3 их переносить нельзя.

4. Дождаться успешной записи и проверки каждого из 4 образов без ошибки.
5. Отключить USB модуля.

На Micro TX V2 из учебного примера запись этих 4 образов при 230400 и проверка хешей успешно выполнены. Если возникает Failed to connect, сначала проверить правильный COM и кабель; не выбирать случайный другой target. Если запись обрывается, сохранить весь лог и повторить после проверки питания/кабеля. [Параметры esptool](#).

[[pagebreak]]

4. Вернуть Lua и проверить TX

Lua - меню управления модулем, хранящееся на SD пульта. Обновление этого файла не заменяет прошивку TX.

1. Скопировать скачанный `elrs.lua` на SD в **SCRIPTS/TOOLS/**.
2. Безопасно извлечь карту из кардридера.
3. Вставить SD в выключенный TX16S.
4. Включить пульт с подключённой антенной внешнего TX.
5. Коротко нажать **MDL** для настроек текущей модели.
6. В **Model Setup** проверить **Internal RF → Off** для этой модели с внешним модулем.
7. Проверить **External RF → CRSF**.
8. Коротко нажать **SYS**.
9. В **Tools** выбрать **ExpressLRS**.
10. Коротко нажать ролик для запуска меню.

Ожидается отклик внешнего TX и версия **4.1.0**, также проверяемая на его OLED. Если меню показывает Loading, сначала проверить External RF/CRSF, питание модуля и что открыт новый Lua. При нескольких одноимённых скриптах старые сохранены в копии SD; убрать их из SCRIPTS/TOOLS, оставив новый. [Установка Lua](#), [подготовка пульта](#).

5. Проверить с подготовленным RX 900 MHz

RX уже должен иметь ELRS 4.x и совместимый регион FCC915. Это проверка TX после замены прошивки; привязывать к нему SuperD 2.4 GHz из основного маршрута нельзя.

1. Включить дрон с совместимым RX, снятыми пропеллерами и выключенным ARM.
2. Проверить связь в Lua и движение каналов в Betaflight → **Receiver**.
3. Если связи нет, в Betaflight 4.5+ нажать **Bind Receiver** на вкладке Receiver.
4. После перевода RX в Bind выбрать **Bind** в меню ExpressLRS пульта.
5. Проверить движение всех нужных каналов.
6. Перезапустить питание дрона.
7. Проверить, что связь восстановилась без повторного Bind.

В проверенном учебном примере был выбран Bind на TX, после чего команды появились в Betaflight. Этот факт подтверждает работоспособность пары; он не доказывает, что у любого RX достаточно нажать только кнопку на TX.

Перед применением ELRS 4.x проверить способ ARM. Традиционный вариант использует CH5/AUX1; новый **Arm using Switch** требует EdgeTX 2.11+ и ELRS 4.x. Отсутствие этого нового пункта само по себе не означает неисправность радиоканала. Версию EdgeTX читают в SYS → VERSION. Наличие движущихся стиков ещё не проверяет ARM. [Настройки ARM для ELRS 4](#).

Откат TX при необходимости

Только для проверенной полной копии этого же модуля. Пульт выключен; USB подключён к внешнему TX. В PowerShell папки esptool задать \$txPort с актуальным COM, затем выполнить:

```
.\esptool.exe -p $txPort -b 115200 write-flash --erase-all 0 tx-original.bin
.\esptool.exe -p $txPort -b 115200 verify-flash 0 tx-original.bin
```

Вторую строку выполнять только после успешного завершения первой. После успешного сравнения отключить USB и включить пульт. Адрес 0 здесь нужен для полной исходной

копии; новый `firmware.bin` приложения так не записывают.

Термины в инструкции

Английские названия сохранены для поиска в интерфейсе; рядом украинский и русский эквиваленты.

English	Українська / русский и смысл
TX / Transmitter	Передавач / передатчик. Радиомодуль пульта.
RX / Receiver	Приймач / приёмник. Принимает команды на дроне.
FC / Flight Controller	Політний контролер / полётный контроллер. Получает команды от RX.
Hardware Target	Цільова плата / целевая плата. Аппаратное описание для прошивки.
Bootloader / BOOT	Завантажувач / загрузчик. Принимает новую прошивку; BOOT выбирает его при включении.
Bind	Прив'язка / привязка. Связывает конкретные TX и RX.
Binding Phrase	Фраза прив'язки / фраза привязки. Задаёт идентификатор пары.
UART	Послідовний інтерфейс / последовательный интерфейс. Проводная связь TX/RX.
TLM / Telemetry	Телеметрія / телеметрия. Обратные данные от RX к пульту.