

# VLT Laser Controller

Лазерный контроллер имеет 4 независимых канала управления: UDP, Веб-интерфейс и Дисплей + Энкодер, SD карта.

Значений по умолчанию для сети (в.ч. после default): веб интерфейс -> <http://192.168.1.1/>, udp -> IP 192.168.1.1, Port 5011, Шлюз 192.168.1.0, Маска подсети 255.255.255.0

Рекомендуем сразу ознакомиться с возможностью веб интерфейса и выполнить необходимые сетевые настройки. HELP прописан в веб интерфейсе. Запомни: введенный UDP порт для устройства – это +1 для порта PC. Например, если указать порт 5011, то порт ответа от устройства на PC будет 5012. Для защиты «забычивости» пользователя последний октет мак адреса автоматически приравнивается к последнему октету IP адреса но в HEX формате (что бы не было с одинаковыми мак адресами нескольких устройств в одной сети).

## Основной функционал:

1. Контроллер работает с форматом ILDA и ILDA Stream (UDP), версии ILDA v5 и v0
2. На сегодня максимальное допустимое количество точек в 1 кадре = до 7000 points (ILDA).
3. Максимальный scanrate до 100 000 pps / Разрешающая способность X-Y-Color 16 бит (65535 позиций x 65535 позиций).
4. Тройная буферизация данных по сети, полноценный Веб интерфейс с поддержкой POST и GET запросов + подтверждение исполнения команд, автономная память для настроек, часы реального времени с планировщиком и автономным питанием (идут даже при отключенном питании от сети, батареи хватит на 3-4 года), высокостабильное питание с диапазоном от 5 до 18 вольт, полноценная сеть с ответами на PING, UDP с поддержкой работы не только по Ethernet но и Internet, наличие уникальных ID девайса (это ID чипа STM - можно привязать лицензию софта например или инфицировать приборы например при настройке через Интернет – это можно будет делать без RDP теперь только порт в роутере клиент наружу должен разрешить)...
5. Важно! Контроллер работает по принципу «передал и забыл», т.е. можно отправить 1 кадр 1 раз и плеер будет его транслировать до тех пор, пока не придёт другой кадр или сетевая команда на остановку трансляции или дисконекта, нет необходимости транслировать 1 и тот же кадр постоянно.

## Организация связи UDP:

Все сетевые команды передаются с размером в 6 байт. `byte[] commandBytes = new byte[6];`

1. Передать команду "LINK1", где LINK – это текст в ASCII, и 1 это байт 0x01; В ответ контроллер пришлет команду "vlt+номер порта на который он будет отвечать по UDP". Этот порт в дальнейшем необходимо прослушивать. Для справки: ответный порт – это +1 к порту соединения. Например «vlt5012» - как только команда пришла, контроллер настроил все необходимые буфера и регистры и готов к работе, а так же система автоматического выключения будет отключена, прибор активируется через 2-3 сек.
2. Передать команду "PLAY10" – это по сути команда разрешающая трансляцию с контроллера. По этому триггеру определяется статус вещания как включен / выключен. Подробнее ниже...
3. Передача ILDA файла (в формате v0 или v5).
  1. Если ILDA содержит больше 1 кадра, то данные нужно передавать покадрово.
  2. Каждый кадр нужно разбить по 1458 байт (ниже MTU). Логика такая, если веб интерфейс включен (глобально) то мы можем передавать до 5-ти пакетов (включительно) по 1458 байт, дальше ждем от контроллера на port PC командное слово "act" – что означает лазерный контроллер готов к приему данных дальше. И передаем пакеты 1 кадра дальше и опять до 5 пакетов... Так же в конце каждого принятого кадра контроллер выполняет подтверждения так же командным словом "act". Например: если 1 кадр весит 8135 байт -> должно быть 5 передач по 1458 -> дальше прием

act -> передача остатка 845 байт -> опять прием act (конец кадра) -> если нужно, можно слать следующий кадр или организовать задержку между кадрами в ms. По такой логике можно передавать до 7000 точек или  $7000 \times 8 + 32$  хедер илда = 56 032 байта.

Если веб интерфейс выключен то за раз можно передавать не по 5, а по 10 пакетов – это ускоряет передачу данных значительно и повышает скорость отработки 1 кадра при потоке кадров (поточковой графике).

Каждый новый кадр система понимает по ключевому слову ILDA находящемуся в хедере...

**Доступные UDP команды (внимание некоторые нужно выполнять совмещённо, например с reset) :**

- Все сетевые команды передаются с размером в 6 байт. `byte[] commandBytes = new byte[6];`
  - 3. **LINK1** - 1 команда на соединение в ответ на которую приходит vlt+номер ответного порта. **LINK0** – команда на дисконект, в ответ на нее приходит подтверждающее слово dis и еще 2 байта (первый показывает включено ли автовыключение лазера после дисконекта 1 включено, 0 выключено и второй байт через сколько минут будет выключен автоматически прибор если автовыключение активно).
  - 4. **PLAY10** или **PLAY01** или **PLAY00**. **PLAY10** – запуск трансляции последнего кадра в памяти или нового приходящего. **PLAY01** – остановка трансляции и очистить буфер в котором хранится последний кадр. **PLAY00** - остановка трансляции и не трогать буфер в котором хранится последний кадр (тогда его можно повторно включить – получается что то наподобие паузы).
  - 5. **SCAN** + 2 байта значения (до 65535) – задежка сканрейта в тактоциклах. Настоящий сканрейт вычислять =  $1000000 / (\text{float})(\text{SCAN} + 2 \text{ байта значения} \times 0.00462962)$ ; В.ч. перевод назад в режим Scanrate
  - 6. **SIZX** + 2 байта значения – размер по X
  - 7. **SIZY** + 2 байта значения – размер по Y
  - 8. **POSX** + 2 байта значения – позиция по X – значения со ЗНАКОМ от -32768 до +32767
  - 9. **POSY** + 2 байта значения – позиция по Y – значения со ЗНАКОМ от -32768 до +32767
  - 10. **SHIF** + 1 байт значение – смещение колор шифт – значения со ЗНАКОМ от -20 до +20
  - 11. **FLIPX** + 1 байт значение – 1 включено, 0 выключено – зеркало по X
  - 12. **FLIPY** + 1 байт значение – 1 включено, 0 выключено – зеркало по Y
  - 13. **ROTA** + 2 байт значение – значение от 0 до 65535 – вращение фигуры – нагрузет проц 😊, лучше держи 0 - выключено
  - 14. **BLANK** + 1 байт значение – 1 бланкинг будет выключен т.е. не модулировать лазер, 0 – бланкинг включен, штатный режим работы
  - 15. **FPSCF** + 1 байт значение – переход на режим от scanrate на FPS – где система сама подстраивает значение scanrate исходя из количества точек в каждом новом кадре пытаюсь удерживать нужный FPS (но заданы варота допустимого минимума и максимума). Значения FPS же можно задавать в диапазоне от 0 до 100.
  - 16. **SHIAV** + 1 байт значение – 1 включено, 0 выключено вручную т.е. через команду SHIF(выше читать) – подключение автоматического колоршифта, работает только если включен режим FPS т.е. кокетто значение с командой FPSCF было передано перед этим...
  - 17. **MRED** + 2 байта значения – максимальная разрешенная яркость красного цвета. От 0 до 65535.
  - 18. **MGRN** + 2 байта значения – максимальная разрешенная яркость зеленого цвета. От 0 до 65535.
  - 19. **MBLU** + 2 байта значения – максимальная разрешенная яркость синего цвета. От 0 до 65535.
- ВНИМАНИЕ! Если два цвета указаны в 0, то прибор сам перестойтся на моноцветный режим, иные цвета будут переведены на него. Например красный 0, синий 0 – прибор станет все цвета отражать зеленым (неважно что в ILDA файле) – но яркость зеленого при этом регулироваться начнет только откоманды MGRN игнорирую значения в ILDA файле- на самом деле это очень удобно.
- 20. **INFO** – по udp вернутся список настроек лазера вида:  
`IP_0=192&IP_1=168&IP_2=6&IP_3=200&port=5011&gate_ip_1=192&gate_ip_2=168&gate_ip_3=6&gate_ip`

\_4=1&mask\_1=255&mask\_2=255&mask\_3=255&mask\_4=0&mac\_1=0x00&mac\_2=0x08&mac\_3=0xDC&mac\_4=0x22&mac\_5=0x4D&mac\_6=0xC8&size\_X=10708&size\_Y=10708&pos\_X=-2785&pos\_Y=11779&rota=0&flip\_X=0&flip\_Y=1&scanrate\_chek=1&scanrate=3855&fps=30&avto\_colorshift=0&colorshift=-3&blanking\_off=0&max\_green=65535&max\_blue=0&max\_red=0&fist\_points=0&last\_points=0&after\_blank\_points=0&before\_blank\_points=0&angle\_points=0&time\_hours=14&time\_minutes=12&auto\_off=0&stop\_work=11&laser\_ID=1105&firmware\_ver=1\_1\_v\_5&stabl=1&laser\_on=0&temp\_galvo\_fist=00&temp\_galvo\_last=0&temp\_laser\_fist=00&temp\_laser\_last=0&web\_server=1&lx=0&cr=0&end=0

ВНИМАНИЕ: можно отслеживать исполнение команд, т.к. данные берутся прямо из памяти... Кстати например cr=0 –ключ аварии лазера или гальво!!!

21. **TEST** + 1 байт значение – от 0 до 5. Где 1 включение встроенного тестировочного кадра GALVO, 2 ILDA Test, 3 Circle, 4 Color, 5 Point, 0 выключить тестирование.
22. **LASPO** + 1 байт значение – от 0 до 255 – добавлять последнюю точку в кадре N раз.
23. **FISPO** + 1 байт значение – от 0 до 255 – добавлять первую точку в кадре N раз.
24. **BLBEF** + 1 байт значение – от 0 до 255 – добавлять точки в кадре N раз, перед каждой линией бланкинга.
25. **BLAFT** + 1 байт значение – от 0 до 255 – добавлять точки в кадре N раз, после каждой линии бланкинга.
26. **ANGPO** + 1 байт значение – от 0 до 255 – добавлять точки в кадре N раз, на каждом угле 90 градусов.

ВНИМАНИЕ! Когда все значения LASPO FISPO BLBEF BLAFT ANGPO = 0, то корректор отключается это добавляет эффективности в скорости, но помогает быстро исправлять явные ошибки при подготовке графики использовать по усмотрению...

27. **TIME** + 2 байта значения – 1й байт часов от 0 до 23, 2й минут от 0 до 59 – настраивает текущее время в контроллере. Часы идут даже при выключенном питании.
28. **ATOF** + 2 байта значения – 1й байт 0 или 1 – это включить или выключить систему автовыключения прибора, 2й байт – это время через сколько выключить (от 0 до 255 мин) прибор после команды дисконект ( т.е. команды LINK0). Лайфхак – ATOF10 – прибор будет выключаться моментально (примерно 1-2 сек) после дисконекта т.к. 0 минут так будет передано 😊
29. **RESET** + 1 байта значения, нужно передавать только RESET1 – лазерный контроллер будет перезагружен физически и прибор в.ч. занимает примерно 3 сек времени , после нужно выполнить процедуру коннект т.е. команда LINK1... и.т.д.
30. **DEFLT** + 1 байта значения, от 1 до 3. 1 – это установка значений лазера по умолчанию (всех но без сброса настроек сети), перезагрузка не нужна. 2 - установка дефолтных значений для сети 192.168.1.1 и порт 5011 + автоматический перезапуск. 3 – это 1 и 2 совмещенный... т.е. сброс всего, в.ч. сети
31. **SAVED** + 1 байта значения, от 1 до 2. 1 – сохранить настройки для сети в автономную память (после перезагрузки прибора они уже не будут сбрасываться) от этой команды прибор перезапустится сам (напоминаю 3 сек на перезапуск). 2 – сохранить ВСЕ!!! кроме настроек для сети в автономную память, перезапуска не будет, он тут не нужен...
32. **WEBSR** + 1 байта значения, от 0 до 1. 1 – вебсервер включен, 0 выключен. После передачи команды для корректной работы выполнить RESET1 (перезагрузка), а вот SAVED выполнять тут не надо, контроллер и так запомнит эту команду...
33. **IP** + 4 байта значения – настройка IP адреса контроллера, пример IP1921686200, каждый из 4х байт это свое значение... затем выполнить SAVED1 (т.е. с перезапуском контроллера)
34. **PORT** + 2 байта значения – номер udp порта контроллера, ответный на ПК это +1 к нему от udp до 65535
35. **GW** + 4 байта значения – все так же как с IP - настройка шлюз
36. **MK** + 4 байта значения – все так же как с IP - настройка маски подсети
37. **QUERY** – контроллер на эту команду отвечает “act”. Данная команда может использоваться для 2х целей: первая, проверка связи (работы) физически контроллера, т.к. ping проверит только работу

сетевого модуля. Вторая: перевод системы автовыключения (если она включена) в другой алгоритм работы, если базово автовыключение срабатывает с момента дисконект, то после передачи **QUERY** автовыключение начнет работать по таймеру с момента последнего **QUERY** и **каждый приход этой команды будет обнулять счетчик времени**. Например: если автовыключение активно и установлено на 5 мин, то при работе через алгоритм **QUERY**, нужно раз в <5 мин передавать команду **QUERY** или **прибор выключится**. Пользоваться этой командой осторожно – не сделай ddos контроллера, т.к. на каждый такой ответ контроллер вынужден прерываться на ответ, что заставляет немного упасть сканрейту (частоте) на пару ms, визуальнo линия (точка в конце) может немного прыгнуть но это на очень короткий промежуток времени (тоже может происходит при передачи других команд – но это совершенно не критично и не мешает). Я мог бы решить эту проблему через RTOS но это сильно снизило бы максимальный потенциал контроллера.

P.s.

1. Конкретно в этой версии контроллера нет дисплея и энкодера, потому сброс на настройки по умолчанию либо через udr либо вытащить и поставить назад батарейку в контроллере.
2. Есть небольшой шум при выводе (он не значительный), но в оригинальной версии стабильность будет в сотню раз выше (т.к. уже изменили принципиальную ошибку).
3. Дополнительно будет добавлена новая команда по udr – высоко стабильное удержание кадра- это когда частота будет следовать без изменений с 6 знаками после запятой (связано с пунктом выше).
4. Если надо можно вводить новые udr команды, например включение . выключения дисплея, перезагрузка модуля сети... все что угодно... по запросу доделаю...