Audion Media Tools - операции

English · Обёртки Scripts

Содержание

- АУДИО - Извлечь, вернуть, конвертировать
- КОДИРОВАНИЕ - Архив, монтаж, хранение, доставка, RAW
- УПАКОВКА - Другой контейнер без перекодирования
- НАРЕЗКА - Оставить, вырезать, разделить
- ЧАСТОТА - Varispeed и conform в 23,976
- ЦВЕТ - LUT, гамма до него, HDR в SDR
- СКАЧАТЬ - Видео или звук по ссылке
- СВЕДЕНИЯ
- СЕРВИС: проверки и установка
- Кнопки у полей
- Что окно делает само

> Собрано из данных программы командой `media_cli.cmd map --docs`: манифест операций (`config\tool_manifest.yaml`), раскладка окна (`config\window_layout.yaml`) и таблица обёрток (`config\wrappers.yaml`). Руками не правится: текст меняется там, и команда запускается снова.

редкое - поле блока «Дополнительно», в конце формы.

АУДИО - Извлечь, вернуть, конвертировать

Запустить аудио - `audio_run`

Обработать аудио выбранным пресетом.

Вызов: `media_cli.cmd run --operation audio_run`

Режимы ленты: ИЗВЛЕЧЬ `audio_workflow=extract`, ЗВУК В ВИДЕО `audio_workflow=in_video`, КОНВЕРТИРОВАТЬ `audio_workflow=audio_files`.

Обёртки: `audio-downmix-51-to-20`, `audio-extract-audio-copy`, `audio-extract-audio-wav`, `audio-extract-audio-wav-32`, `audio-lufs-14`, `audio-lufs-23`, `audio-resample-to-44100`, `audio-resample-to-44100-32f`, `audio-resample-to-44100-32f_sync`, `audio-resample-to-44100_sync`, `audio-resample-to-48000`, `audio-resample-to-48000-32f`, `audio-resample-to-48000-32f_sync`, `audio-resample-to-48000_sync`, `audio-to-flac`, `audio-to-flac-48k`, `audio-to-flac-96k`, `audio-to-m4a-256`, `audio-to-m4a-384`, `ff-extract-audio`.

Пресеты - Режим работы `audio_profile`

Пресет	Что это	Что ставит
`copy_auto` Copy	Скопировать выбранный аудиопоток без перекодирования; Auto выберет M4A/MKA/родной файл по кодеку и каналам.	audio_stream_mode=first, audio_format=copy, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=auto, audio_copy_container=auto, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`remux_container` Remux	Скопировать аудио в выбранный аудиоконтейнер без перекодирования; удобно для AAC -> M4A или многоканала -> MKA.	audio_stream_mode=first, audio_format=copy, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=auto, audio_copy_container=mka, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`wav_24_48` WAV 24/48	Монтажный PCM WAV без сжатия.	audio_stream_mode=first, audio_format=wav, audio_sample_rate=48000, audio_bit_depth=24, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mov, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`wav_32f_48` WAV 32f/48	WAV 32-bit float для реставрации и дальнейшей обработки.	audio_stream_mode=first, audio_format=wav, audio_sample_rate=48000, audio_bit_depth=32f, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mov, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`flac_24_48` FLAC 24/48	Аудио без потерь для архива.	audio_stream_mode=first, audio_format=flac, audio_sample_rate=48000, audio_bit_depth=24, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mkv, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`alac_m4a` ALAC M4A	Apple Lossless внутри M4A: без потерь, но компактнее WAV.	audio_stream_mode=first, audio_format=alac, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=auto, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off

Пресет	Что это	Что ставит
`m4a384` M4A 384k	AAC 384k для просмотра или публикации.	audio_stream_mode=first, audio_format=m4a, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`m4a256` M4A 256k	Компактный AAC 256k для заливки и портативного прослушивания.	audio_stream_mode=first, audio_format=m4a, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=256k, audio_resampler=soxr, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`video_pcm24_48_soxr` Video PCM	Не трогать видеопоток, пересэмплировать аудио внутри MOV.	audio_workflow=in_video, audio_stream_mode=first, audio_format=wav, audio_sample_rate=48000, audio_bit_depth=24, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mov, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`video_aac384_48_soxr` Video AAC	Не трогать видеопоток, перекодировать аудио внутри контейнера.	audio_workflow=in_video, audio_stream_mode=first, audio_format=m4a, audio_sample_rate=48000, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`mp3_v0` MP3 V0	Самый качественный LAME VBR профиль, около 245 kbps.	audio_stream_mode=first, audio_format=mp3, audio_sample_rate=48000, audio_bit_depth=source, audio_bitrate=320k, audio_resampler=soxr, audio_video_container=source, mp3_lame_preset=v0, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`mp3_insane` MP3 320	LAME CBR 320 kbps.	audio_stream_mode=first, audio_format=mp3, audio_sample_rate=48000, audio_bit_depth=source, audio_bitrate=320k, audio_resampler=soxr, audio_video_container=source, mp3_lame_preset=insane, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`opus192` Opus 192k	Современный компактный формат для речи и музыки.	audio_stream_mode=first, audio_format=opus, audio_sample_rate=48000, audio_bit_depth=source, audio_bitrate=192k, audio_resampler=soxr, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off

Пресет	Что это	Что ставит
`lufs14_m4a` LUFS -14 M4A	Двухпроходная нормализация под распространённую streaming loudness-цель.	audio_stream_mode=first, audio_format=m4a, audio_sample_rate=48000, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=two_pass, audio_lufs=-14
`downmix_wav` Stereo WAV	Свести первый аудиопоток в стерео.	audio_stream_mode=first, audio_format=wav, audio_sample_rate=48000, audio_bit_depth=24, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mov, audio_channel_mode=stereo, audio_lufs_mode=off, audio_lufs=off
`matrix51_wav` Matrix 5.1	Стерео-совместимый DPLII matrix downmix.	audio_stream_mode=first, audio_format=wav, audio_sample_rate=48000, audio_bit_depth=24, audio_bitrate=384k, audio_resampler=soxr, audio_video_container=mov, audio_channel_mode=stereo, audio_lufs_mode=off, audio_lufs=off, matrix_51=true
`copy_all_audio` Copy all	Извлечь каждый аудиопоток без перекодирования.	audio_stream_mode=all, audio_format=copy, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=auto, audio_copy_container=auto, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=off, audio_lufs=off
`lufs_report` LUFS отчёт	Проанализировать громкость и записать отчёт без кодирования.	audio_stream_mode=first, audio_format=m4a, audio_sample_rate=source, audio_bit_depth=source, audio_bitrate=384k, audio_resampler=auto, audio_video_container=source, audio_channel_mode=keep, audio_lufs_mode=report, audio_lufs=-23

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`wav` WAV; `w64` W64; `flac` FLAC; `m4a` M4A; `m4b` M4B; `mp3` MP3; `aac` AAC; `ac3` AC3; `eac3` E-AC3; `dts` DTS; `opus` Opus; `oga` OGA; `ogg` OGG; `aif` AIF; `aiff` AIFF; `caf` CAF; `au` AU; `snd` SND; `mka` MKA; `ape` APE; `wv` WV; `tak` TAK; `tta` TTA; `mpc` MPC; `amr` AMR; `awb` AWB; `spx` SPX; `dsf` DSF; `dff` DFF; `mp4` MP4; `m4v` M4V; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mpg` MPG; `mpeg` MPEG; `mts` MTS; `m2ts` M2TS; `ts` TS; `3gp` 3GP	-	Источник обходится рекурсивно; Назначение сохраняет такую же структуру подпапок.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Формат `audio_format`	выбор	`wav`	`copy` Copy; `wav` WAV; `flac` FLAC; `alac` ALAC; `m4a` M4A / AAC; `mp3` MP3; `opus` Opus	-	Формат результата. Copy сохраняет выбранный аудиопоток без перекодирования; ALAC пишет Apple Lossless в M4A.
Контейнер Copy `audio_copy_container`	выбор	`auto`	`auto` Auto; `m4a` M4A; `mka` MKA; `elementary` Файл кодека	видно, когда все из (audio_format = copy; audio_workflow - одно из extract, audio_files)	Политика контейнера для извлечённых Copy-файлов: Auto выбирает по кодеку и числу каналов; MKA/M4A/файл кодека задают упаковку вручную.
Частота дискретизации `audio_sample_rate`	выбор	`48000`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	скрыто, когда audio_format = copy	Целевая частота дискретизации. Родная оставляет исходную частоту.
Разрядность `audio_bit_depth`	выбор	`24`	`source` COPY; `16` 16-bit; `24` 24-bit; `32f` 32-bit float	скрыто, когда audio_format = copy	Разрядность PCM/FLAC. Родная сохраняет исходную разрядность, если формат это поддерживает.
Ресемплер `audio_resampler`	выбор	`soxr`	`auto` Copy — сохранить исходное (видно, когда audio_format = copy); `soxr` SoX 100% quality	скрыто, когда audio_format = copy	Движок ресемплинга при изменении частоты дискретизации.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Выровнять метки времени (редкое) `audio_resync`	флажок	`false`	-	скрыто, когда audio_format = copy	Заполняет пропуски звука и срезает наложения по меткам времени, от нуля (aresample=async=1:first_pts=0): запись с битыми метками не расходится с картинкой.
Контейнер видео `audio_video_container`	выбор	`source`	`source` Исходный; `mov` MOV (недоступно, когда audio_format - одно из flac, opus); `mkv` MKV; `mp4` MP4 (недоступно, когда audio_format - одно из wav, flac); `mxf` MXF (недоступно, когда audio_format - одно из flac, alac, m4a, mp3, opus)	видно, когда audio_workflow = in_video	Контейнер для режима обновления звука в видео. Небезопасные пары кодек/ контейнер отключаются.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_format - одно из m4a, opus; все из (audio_format = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Аудиопотоки `audio_stream_mode`	выбор	`first`	`first` Первый поток; `all` Все потоки; `index` По индексу; `language` По языку	-	Какие аудиопотоки брать из каждого исходного файла.
Индекс потока `audio_stream_index`	число	`0`	-	видно, когда audio_stream_mode = index	Индекс аудиопотока с нуля: a0 означает первый аудиопоток.
Язык `audio_language`	выбор	-	`""` Не выбран; `eng` Английский (eng); `ara` Арабский (ara); `spa` Испанский (spa); `ita` Итальянский (ita); `zho` Китайский (zho); `kor` Корейский (kor); `deu` Немецкий (deu); `pol` Польский (pol); `por` Португальский (por); `rus` Русский (rus); `ukr` Украинский (ukr); `fra` Французский (fra); `jpn` Японский (jpn)	видно, когда audio_stream_mode = language	Выбор дорожек по языковому тегу контейнера, например eng, rus или ukr.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
LUFS `audio_lufs`	выбор	`off`	`off` Выкл.; `-14` -14; `-16` -16; `-23` -23	видно, когда все из (audio_lufs_mode - одно из report, one_pass, two_pass; не (audio_format = copy))	Целевая интегральная громкость для loudnorm.
Режим LUFS `audio_lufs_mode`	выбор	`off`	`off` Выкл.; `report` Отчёт; `one_pass` One pass; `two_pass` Two pass	скрыто, когда audio_format = copy	Отчёт только измеряет громкость; один проход быстрее; два прохода точнее.
Каналы `audio_channel_mode`	выбор	`keep`	`keep` Сохранить; `stereo` Stereo; `mono` Mono; `dual_mono` Dual mono	скрыто, когда audio_format = copy	Операция с каналами: сохранить, свести в stereo/mono или сделать dual mono.
5.1 Matrix `matrix_51`	флажок	`false`	-	видно, когда все из (audio_channel_mode = stereo; не (audio_format = copy))	Добавляет DPLII matrix-совместимость при сведении в стерео.
Сведение `downmix_formula`	выбор	`ffmpeg`	`ffmpeg` FFmpeg; `surround_6db` Тылы -6 дБ	видно, когда все из (audio_channel_mode = stereo; matrix_51 выключено; не (audio_format = copy))	Как 5.1 становится стерео. Стерео-источник в обоих случаях остаётся как есть.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_format = mp3	Режим качества LAME для MP3.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

КОДИРОВАНИЕ - Архив, монтаж, хранение, доставка, RAW

Создать архив - `archive_run`

Кодировать выбранные архивные форматы.

Вызов: `media_cli.cmd run --operation archive_run`

Пресеты - Архивный пресет `archive_profile`

Пресет	Что это	Что ставит
`ffv1_mkv` FFV1 MKV + FLAC	Архивный мастер со всеми потоками и FLAC аудио; PCM с плавающей точкой, которого FLAC не держит, копируется как есть. Пиксельный формат, который FFV1 не сохранит, отклоняется.	target_codecs=ffv1, audio_mode=flac, output_container=mkv, pix_fmt=auto
`x264_lossless_mkv` x264 lossless MKV	CRF 0 x264 как архивный вариант, когда FFV1 слишком тяжелый. Держит YUV 8 и 10 бит; источник 12 бит, RGB или с альфой отклоняется — его сохранит FFV1.	target_codecs=x264_lossless, audio_mode=source, output_container=mkv, pix_fmt=auto
`dual_archive` FFV1 + x264 lossless	Последовательно создать оба архивных формата.	target_codecs=ffv1,x264_lossless, audio_mode=flac, output_container=mkv, pix_fmt=auto

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Архивные форматы `target_codecs`	флажки	-	`ffv1` FFV1; `x264_lossless` x264 lossless	-	Один или несколько выходных таргетов. Каждый таргет пишется отдельно для каждого файла Источника.
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.
Аудио `audio_mode`	выбор	`source`	`source` COPY; `aac_320` AAC override; `mp3` MP3; `flac` FLAC; `pcm_s16` PCM 16-bit; `pcm_s24` PCM 24-bit; `pcm_f32` PCM float; `none` Без аудио	-	Политика аудио результата: copy потока, lossy/lossless-кодирование, разрядность PCM или без аудио. Для MXF доступны только совместимые PCM-режимы.
Частота дискретизации `audio_sample_rate`	выбор	`source`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	видно, когда audio_mode - одно из aac_320, mp3, flac, pcm_s16, pcm_s24, pcm_f32	Целевая частота дискретизации. Родная оставляет исходную частоту.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_mode = mp3	Режим качества LAME для MP3.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_mode = aac_320; все из (audio_mode = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).

Поле	Вид	По умолчанию	Варианты	Когда	Что это
CPU preset `cpu_encoder_preset`	выбор	`medium`	`fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	скрыто, когда encode_backend - одно из cuda, qsv, amd	Software x264/x265 preset скорости/сжатия. Более медленные presets тратят больше CPU на более чистое/компактное кодирование при том же CRF.
Pixel format `pix_fmt`	выбор	`auto`	`auto` auto; `yuv420p` yuv420p; `yuv420p10le` yuv420p10le; `yuv422p` yuv422p; `yuv422p10le` yuv422p10le	-	Pixel format и разрядность результата. Значение должно соответствовать выбранному кодеку и монтажному либо delivery-сценарию.
Тюнинг кодирования `encode_tuning`	флажки	-	`grain` Grain; `film` Film; `animation` Animation; `fastdecode` Fast decode; `zerolatency` Zero latency	скрыто, когда encode_backend - одно из cuda, qsv, amd	Значения tune для x264/x265. Аппаратные кодеры используют свои регуляторы качества и не читают этот список, поэтому он доступен только для CPU.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

Создать монтажные файлы - `editing_run`

Кодировать выбранные монтажные цели.

Вызов: `media_cli.cmd run --operation editing_run`

Обёртки: `ff-dnxhr-hq-mov`, `ff-dnxhr-hq-mxf`, `ff-dnxhr-hqx-mov`, `ff-dnxhr-hqx-mxf`, `ff-prores-422-mov`, `ff-prores-422-mxf`, `ff-prores-lt-mov`, `ff-prores-lt-mxf`, `ff-prores-proxy-mov`, `ff-prores-proxy-mxf`.

Пресеты - Монтажный пресет `editing_profile`

Пресет	Что это	Что ставит
`prores_proxy_mov` Proxy	Лёгкий proxy для отзывчивого монтажа.	target_codecs=prores_proxy, output_container=mov, audio_mode=pcm_s16, pix_fmt=yuv422p10le
`prores_lt_mov` LT	Облегчённый монтажный ProRes.	target_codecs=prores_lt, output_container=mov, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`prores_422_mov` 422	Сбалансированный 10-bit 4:2:2 mezzanine.	target_codecs=prores_422, output_container=mov, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`prores_hq_mov` HQ	Высокобитрейтный ProRes 422 HQ.	target_codecs=prores_hq, output_container=mov, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`prores_proxy_mxf` Proxy	ProRes Proxy в MXF с PCM 16-bit.	target_codecs=prores_proxy, output_container=mxf, audio_mode=pcm_s16, pix_fmt=yuv422p10le
`prores_lt_mxf` LT	ProRes LT в MXF с PCM 24-bit.	target_codecs=prores_lt, output_container=mxf, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`prores_422_mxf` 422	ProRes 422 в MXF с PCM 24-bit.	target_codecs=prores_422, output_container=mxf, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`prores_hq_mxf` HQ	ProRes 422 HQ в MXF с PCM 24-bit.	target_codecs=prores_hq, output_container=mxf, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`dnxhr_hq_mov` HQ	DNxHR HQ 8-bit в MOV.	target_codecs=dnxhr_hq, output_container=mov, audio_mode=pcm_s24, pix_fmt=yuv422p
`dnxhr_hqx_mov` HQX	DNxHR HQX 10-bit в MOV.	target_codecs=dnxhr_hqx, output_container=mov, audio_mode=pcm_s24, pix_fmt=yuv422p10le
`dnxhr_hq_mxf` HQ	DNxHR HQ 8-bit в MXF с PCM 24-bit.	target_codecs=dnxhr_hq, output_container=mxf, audio_mode=pcm_s24, pix_fmt=yuv422p
`dnxhr_hqx_mxf` HQX	DNxHR HQX 10-bit в MXF с PCM 24-bit.	target_codecs=dnxhr_hqx, output_container=mxf, audio_mode=pcm_s24, pix_fmt=yuv422p10le

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.
Монтажные цели `target_codecs`	флажки	-	`prores_proxy` ProRes Proxy; `prores_lt` ProRes LT; `prores_422` ProRes 422; `prores_hq` ProRes HQ; `dnxhr_hq` DNxHR HQ; `dnxhr_hqx` DNxHR HQX	-	Один или несколько выходных таргетов. Каждый таргет пишется отдельно для каждого файла Источника.
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.
Аудио `audio_mode`	выбор	`source`	`source` COPY (недоступно, когда output_container = mxf); `aac_320` AAC override (недоступно, когда output_container = mxf); `mp3` MP3 (недоступно, когда output_container = mxf); `flac` FLAC (недоступно, когда output_container - одно из mxf, mov); `pcm_s16` PCM 16-bit; `pcm_s24` PCM 24-bit; `pcm_f32` PCM float (недоступно, когда output_container = mxf); `none` Без аудио	-	Политика аудио результата: copy потока, lossy/lossless-кодирование, разрядность PCM или без аудио. Для MXF доступны только совместимые PCM-режимы.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Частота дискретизации `audio_sample_rate`	выбор	`source`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	видно, когда audio_mode - одно из aac_320, mp3, flac, pcm_s16, pcm_s24, pcm_f32	Целевая частота дискретизации. Родная оставляет исходную частоту.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_mode = mp3	Режим качества LAME для MP3.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_mode = aac_320; все из (audio_mode = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).
Контейнер `output_container`	выбор	`mov`	`mov` MOV; `mxf` MXF	-	Контейнер результата, если выбранное семейство кодеков его допускает.
Разрешение `output_resolution`	выбор	`source`	`source` Исходное; `2160` 2160p; `1440` 1440p; `1080` 1080p; `720` 720p; `480` 480p	-	Необязательное масштабирование. Исходное сохраняет размер кадра.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Чересстрочный источник (редкое) `scan_mode`	выбор	`double`	`keep` Оставить поля; `progressive` Прогрессив; `double` Прогрессив x2	-	Что станет с источником из двух полей (25i, 29,97i, 50i). «Прогрессив x2», по умолчанию: bwdif делает кадр из каждого поля, частота вдвое выше, движение целиком - 50i становится 50p. «Прогрессив»: цельный кадр из каждого кадра, частота прежняя. «Оставить поля»: поля и их порядок сохраняются - для чересстрочного эфира. Прогрессивный источник не трогается.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

Кодировать хранение - `storage_run`

Кодировать выбранные цели хранения.

Вызов: `media_cli.cmd run --operation storage_run`

Обёртки: `ff-av1-crf14`, `ff-av1-fast-crf14`, `ff-hevc10-crf14`, `ff-x264-4k-to-1080-crf14`, `ff-x264-4k-to-1080-crf17`, `ff-x264-4k-to-1080-crf21`, `ff-x264-crf14`, `ff-x264-crf17`, `ff-x264-crf21`.

Пресеты - Пресет хранения `storage_profile`

Пресет	Что это	Что ставит
`x264_crf14` x264 CRF 14	Универсальное качественное H.264-хранение.	target_codecs=x264, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=source, audio_mode=source, audio_bitrate=384k, crf=14, cpu_encoder_preset=slow, pix_fmt=yuv420p
`x264_1080_crf14` x264 1080p CRF 14	Даунскейл 4K в качественный 1080p.	target_codecs=x264, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=1080, audio_mode=aac_320, audio_bitrate=384k, crf=14, cpu_encoder_preset=slow, pix_fmt=yuv420p
`hevc10_crf14` HEVC Main10 CRF 14	10-bit HEVC для меньших файлов при хорошем качестве.	target_codecs=hevc_x265, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=source, audio_mode=source, audio_bitrate=384k, crf=14, cpu_encoder_preset=slow, pix_fmt=yuv420p10le
`svt_av1_crf14` SVT-AV1 CRF 14	Современный кодек хранения, если поддержка декодирования подходит.	target_codecs=svt_av1, encode_backend=cpu, decode_backend=auto, output_container=mkv, output_resolution=source, audio_mode=source, audio_bitrate=384k, crf=14, av1_preset=6, pix_fmt=yuv420p10le

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.
Кодеки хранения `target_codecs`	флажки	-	`x264` x264 / H.264; `hevc_x265` x265 / HEVC; `svt_av1` AV1	-	Один или несколько выходных таргетов. Каждый таргет пишется отдельно для каждого файла Источника.
Кодер `encode_backend`	выбор	`cpu`	`cpu` CPU; `cuda` NVENC; `amd` AMF; `qsv` QSV	-	Выбор backend кодирования. Ниже показываются только родные параметры выбранного backend.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.
Аудио `audio_mode`	выбор	`source`	`source` COPY; `aac_320` AAC override; `mp3` MP3; `flac` FLAC; `pcm_s16` PCM 16-bit; `pcm_s24` PCM 24-bit; `pcm_f32` PCM float; `none` Без аудио	-	Политика аудио результата: copy потока, lossy/lossless-кодирование, разрядность PCM или без аудио. Для MXF доступны только совместимые PCM-режимы.
Частота дискретизации `audio_sample_rate`	выбор	`source`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	видно, когда audio_mode - одно из aac_320, mp3, flac, pcm_s16, pcm_s24, pcm_f32	Целевая частота дискретизации. Родная оставляет исходную частоту.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_mode = mp3	Режим качества LAME для MP3.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_mode = aac_320; все из (audio_mode = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Контейнер `output_container`	выбор	`mp4`	`mp4` MP4; `mkv` MKV; `mov` MOV	-	Контейнер результата, если выбранное семейство кодеков его допускает.
CPU preset `cpu_encoder_preset`	выбор	`medium`	`fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	скрыто, когда encode_backend - одно из cuda, qsv, amd	Software x264/x265 preset скорости/сжатия. Более медленные presets тратят больше CPU на более чистое/компактное кодирование при том же CRF.
NVENC preset `nvenc_preset`	выбор	`p6`	`p5` NVENC p5; `p6` NVENC p6; `p7` NVENC p7	видно, когда encode_backend = cuda	NVIDIA NVENC P-preset. Это шкала аппаратного кодера, не эквивалент CPU slow/slower/veryslow.
QSV preset `qsv_encoder_preset`	выбор	`medium`	`veryfast` veryfast; `faster` faster; `fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	видно, когда encode_backend = qsv	Intel QuickSync preset. У QSV своя FFmpeg-линейка: veryfast, faster, fast, medium, slow, slower, veryslow.
AMF quality `amf_quality`	выбор	`quality`	`speed` speed; `balanced` balanced; `quality` quality	видно, когда encode_backend = amd	AMD AMF quality preset. Управляет AMF режимом speed/balanced/quality; CQ/QP по-прежнему задаёт уровень квантования.
Разрешение `output_resolution`	выбор	`source`	`source` Исходное; `2160` 2160p; `1440` 1440p; `1080` 1080p; `720` 720p; `480` 480p	-	Необязательное масштабирование. Исходное сохраняет размер кадра.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Чересстрочный источник (редкое) `scan_mode`	выбор	`double`	`keep` Оставить поля; `progressive` Прогрессив; `double` Прогрессив x2	-	Что станет с источником из двух полей (25i, 29,97i, 50i). «Прогрессив x2», по умолчанию: bwdif делает кадр из каждого поля, частота вдвое выше, движение целиком - 50i становится 50p. «Прогрессив»: цельный кадр из каждого кадра, частота прежняя. «Оставить поля»: поля и их порядок сохраняются - для чересстрочного эфира. Прогрессивный источник не трогается.
SVT-AV1 preset `av1_preset`	выбор	`6`	`4` 4 quality; `6` 6 balanced; `8` 8 fast	видно, когда target_codecs - одно из svt_av1; скрыто, когда encode_backend - одно из cuda, qsv, amd	Preset скорости/качества software SVT-AV1. Аппаратные AV1-кодеры используют свои шкалы NVENC/QSV/AMF.
Pixel format `pix_fmt`	выбор	`auto`	`auto` auto; `yuv420p` yuv420p; `yuv420p10le` yuv420p10le; `yuv422p` yuv422p; `yuv422p10le` yuv422p10le	-	Pixel format и разрядность результата. Значение должно соответствовать выбранному кодеку и монтажному либо delivery-сценарию.
CRF `crf`	число	`14`	-	скрыто, когда encode_backend - одно из cuda, qsv, amd	Цель качества для software-кодеков: ниже означает крупнее и чище.
CQ/QP `cq`	число	`14`	-	видно, когда encode_backend - одно из cuda, qsv, amd	Constant Quality для аппаратных кодировщиков.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Тюнинг кодирования `encode_tuning`	флажки	-	`grain` Grain; `film` Film; `animation` Animation; `fastdecode` Fast decode; `zerolatency` Zero latency	скрыто, когда encode_backend - одно из cuda, qsv, amd	Значения tune для x264/x265. Аппаратные кодеры используют свои регуляторы качества и не читают этот список, поэтому он доступен только для CPU.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

Кодировать публикацию - `delivery_run`

Кодировать выбранные цели публикации.

Вызов: `media_cli.cmd run --operation delivery_run`

Пресеты - Пресет публикации `delivery_profile`

Пресет	Что это	Что ставит
`review_h264` Review H.264	Быстрый MP4 для просмотра и отправки.	target_codecs=x264, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=1080, audio_mode=aac_320, audio_bitrate=384k, crf=14, cpu_encoder_preset=fast, pix_fmt=yuv420p
`upload_h264` Upload H.264	Более чистый H.264 MP4 для видеоплатформ.	target_codecs=x264, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=source, audio_mode=aac_320, audio_bitrate=384k, crf=14, cpu_encoder_preset=slow, pix_fmt=yuv420p

Пресет	Что это	Что ставит
`upload_hevc10` Upload HEVC 10-bit	HEVC Main10 для платформ, которые его принимают.	target_codecs=hevc_x265, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=source, audio_mode=aac_320, audio_bitrate=384k, crf=14, cpu_encoder_preset=slow, pix_fmt=yuv420p10le
`svt_av1_delivery` SVT-AV1 delivery	AV1-кодирование для современных публикаций и тестов.	target_codecs=svt_av1, encode_backend=cpu, decode_backend=auto, output_container=mp4, output_resolution=1080, audio_mode=aac_320, audio_bitrate=384k, crf=14, av1_preset=8, pix_fmt=yuv420p10le

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.
Кодеки публикации `target_codecs`	флажки	-	`x264` x264 / H.264; `hevc_x265` x265 / HEVC; `svt_av1` AV1	-	Один или несколько выходных таргетов. Каждый таргет пишется отдельно для каждого файла Источника.
Кодер `encode_backend`	выбор	`cpu`	`cpu` CPU; `cuda` NVENC; `amd` AMF; `qsv` QSV	-	Выбор backend кодирования. Ниже показываются только родные параметры выбранного backend.
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Аудио `audio_mode`	выбор	`source`	`source` COPY; `aac_320` AAC override; `mp3` MP3; `flac` FLAC; `pcm_s16` PCM 16-bit; `pcm_s24` PCM 24-bit; `pcm_f32` PCM float; `none` Без аудио	-	Политика аудио результата: copy потока, lossy/lossless-кодирование, разрядность PCM или без аудио. Для MXF доступны только совместимые PCM-режимы.
Частота дискретизации `audio_sample_rate`	выбор	`source`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	видно, когда audio_mode - одно из aac_320, mp3, flac, pcm_s16, pcm_s24, pcm_f32	Целевая частота дискретизации. Родная оставляет исходную частоту.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_mode = mp3	Режим качества LAME для MP3.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_mode = aac_320; все из (audio_mode = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).
Контейнер `output_container`	выбор	`mp4`	`mp4` MP4; `mkv` MKV; `mov` MOV	-	Контейнер результата, если выбранное семейство кодеков его допускает.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
CPU preset `cpu_encoder_preset`	выбор	`medium`	`fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	скрыто, когда encode_backend - одно из cuda, qsv, amd	Software x264/x265 preset скорости/сжатия. Более медленные presets тратят больше CPU на более чистое/компактное кодирование при том же CRF.
NVENC preset `nvenc_preset`	выбор	`p6`	`p5` NVENC p5; `p6` NVENC p6; `p7` NVENC p7	видно, когда encode_backend = cuda	NVIDIA NVENC P-preset. Это шкала аппаратного кодера, не эквивалент CPU slow/slower/veryslow.
QSV preset `qsv_encoder_preset`	выбор	`medium`	`veryfast` veryfast; `faster` faster; `fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	видно, когда encode_backend = qsv	Intel QuickSync preset. У QSV своя FFmpeg-линейка: veryfast, faster, fast, medium, slow, slower, veryslow.
AMF quality `amf_quality`	выбор	`quality`	`speed` speed; `balanced` balanced; `quality` quality	видно, когда encode_backend = amd	AMD AMF quality preset. Управляет AMF режимом speed/balanced/quality; CQ/QP по-прежнему задаёт уровень квантования.
Разрешение `output_resolution`	выбор	`source`	`source` Исходное; `2160` 2160p; `1440` 1440p; `1080` 1080p; `720` 720p; `480` 480p	-	Необязательное масштабирование. Исходное сохраняет размер кадра.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Чересстрочный источник (редкое) `scan_mode`	выбор	`double`	`keep` Оставить поля; `progressive` Прогрессив; `double` Прогрессив x2	-	Что станет с источником из двух полей (25i, 29,97i, 50i). «Прогрессив x2», по умолчанию: bwdif делает кадр из каждого поля, частота вдвое выше, движение целиком - 50i становится 50p. «Прогрессив»: цельный кадр из каждого кадра, частота прежняя. «Оставить поля»: поля и их порядок сохраняются - для чересстрочного эфира. Прогрессивный источник не трогается.
SVT-AV1 preset `av1_preset`	выбор	`6`	`4` 4 quality; `6` 6 balanced; `8` 8 fast	видно, когда target_codecs - одно из svt_av1; скрыто, когда encode_backend - одно из cuda, qsv, amd	Preset скорости/качества software SVT-AV1. Аппаратные AV1-кодеры используют свои шкалы NVENC/QSV/AMF.
Pixel format `pix_fmt`	выбор	`auto`	`auto` auto; `yuv420p` yuv420p; `yuv420p10le` yuv420p10le; `yuv422p` yuv422p; `yuv422p10le` yuv422p10le	-	Pixel format и разрядность результата. Значение должно соответствовать выбранному кодеку и монтажному либо delivery-сценарию.
CRF `crf`	число	`14`	-	скрыто, когда encode_backend - одно из cuda, qsv, amd	Цель качества для software-кодеков: ниже означает крупнее и чище.
CQ/QP `cq`	число	`14`	-	видно, когда encode_backend - одно из cuda, qsv, amd	Constant Quality для аппаратных кодировщиков.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Тюнинг кодирования `encode_tuning`	флажки	-	`grain` Grain; `film` Film; `animation` Animation; `fastdecode` Fast decode; `zerolatency` Zero latency	скрыто, когда encode_backend - одно из cuda, qsv, amd	Значения tune для x264/x265. Аппаратные кодеры используют свои регуляторы качества и не читают этот список, поэтому он доступен только для CPU.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

Проявить ProRes RAW - `raw_develop_run`

Проявить каждый ProRes RAW Источника на видеокарте в выбранный формат.

Вызов: `media_cli.cmd run --operation raw_develop_run`

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Проявить в `raw_target`	выбор	`prores4444`	`prores4444` ProRes 4444; `dnxhr444` DNxHR 444; `h264` H.264; `h265` H.265 10 бит	-	Во что пишется проявленный клип. ProRes 4444 и DNxHR 444 - 10 бит 4:4:4 для монтажа, со звуком PCM источника; H.264 и H.265 10 бит - файл посмотреть и отправить, звук AAC.
Кодер `raw_encoder`	выбор	`cpu`	`cpu` CPU; `nvenc` NVENC	видно, когда raw_target - одно из h264, h265	Кодер H.264 и H.265: процессор (x264, x265, CRF 16) или NVENC видеокарты NVIDIA (p6, CQ 16).

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Кусков сразу `raw_workers`	выбор	`auto`	`auto` Авто; `2` 2; `4` 4; `8` 8; `12` 12	-	Сколько кусков клипа проявляются одновременно, каждый своим cineraw: декодирование ProRes RAW занимает по ядру на кусок, видеокарта проявляет кадр за миллисекунды. Авто - половина ядер, не больше восьми.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

УПАКОВКА - Другой контейнер без перекодирования

Запустить remux - `remux_run`

Выполнить выбранную упаковку.

Вызов: `media_cli.cmd run --operation remux_run`

Режимы ленты: ВИДЕО `remux_stream_mode=video`, АУДИО `remux_stream_mode=audio`, ВИДЕО И АУДИО `remux_stream_mode=video_audio`.

Обёртки: `ff-remux-mkv-to-mp4`, `ff-remux-mkv-to-mp4-faststart`, `ff-remux-mov-to-mxf-copy`, `ff-remux-mp4-to-mkv`, `ff-remux-mp4-to-mkv-copy`, `ff-remux-mxf-to-mov-copy`.

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Контейнер вывода `remux_target_format`	контейнер	`mp4`	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF	видно, когда remux_stream_mode - одно из video, video_audio	Новый контейнер. Файл, кодек которого контейнер не примет, пропускается с причиной - проверка смотрит на кодек, а не только на контейнер.
Как разворачивать `remux_orientation`	выбор	`none`	`none` Как снято; `cw90` 90 по часовой; `ccw90` 90 против часовой; `turn180` Перевернуть; `hflip` Зеркало по горизонтали; `vflip` Зеркало по вертикали	видно, когда remux_stream_mode - одно из video, video_audio	Записывает в контейнер, как показывать картинку. Ничего не декодируется и не перекодируется — пиксели копируются, меняется одно поле в заголовке. Предлагается для камерных H.264 и HEVC; MP4, MOV и MKV это хранят, MXF нет — и раздел откажет, а не выдаст неповёрнутый файл.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Аудиоконтейнер вывода `remux_audio_target_format`	контейнер	`mka`	`m4a` M4A; `alac` ALAC; `mka` MKA; `aac` AAC; `ac3` AC3; `eac3` EAC3; `dts` DTS; `flac` FLAC; `wav` WAV; `mp3` MP3; `opus` OPUS; `ogg` OGG	видно, когда remux_stream_mode = audio	Файл, в который уходит звук без перекодирования. MKA принимает любой звук, остальные - свои кодеки: M4A - AAC и ALAC, WAV - PCM.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

Вынуть субтитры - `subtitles_extract`

Все потоки субтитров видео, скопированные как есть в файл MKS с его именем; видео без субтитров пропускается.

Вызов: `media_cli.cmd run --operation subtitles_extract`

Обёртки: `ff-extract-subs`.

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

НАРЕЗКА - Оставить, вырезать, разделить

Обрезать файл - `trim_run`

Резать по ближайшему ключевому кадру, копировать потоки, сдвинуть таймкод; звук — по сэмплу.

Вызов: `media_cli.cmd run --operation trim_run`

Режимы ленты: ОСТАВИТЬ `trim_action=keep`, ВЫРЕЗАТЬ `trim_action=cut`, РАЗДЕЛИТЬ `trim_action=split`.

Обёртки: `ff-trim-both`, `ff-trim-end`, `ff-trim-middle`, `ff-trim-start`.

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Какой файл резать `trim_file`	trim_file_pick	-	-	-	Видео- и звуковые файлы из Источника по алфавиту. Режется один файл за раз: у дублей редко совпадают ключевые кадры, не говоря о точках реза.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Пресет разделения `trim_split_mode`	выбор	`points`	`points` По точкам; `duration` Равные части	видно, когда trim_action = split	Задать точки вручную или автоматически разделить на равные части. Части нумеруются от начала файла, остаток идёт последним.
Длительность части `trim_split_duration`	таймкод	`01:29:00,000`	-	видно, когда все из (trim_action = split; trim_split_mode = duration)	Часы, минуты, секунды и миллисекунды, как у точки: 01:29:00,000. Цифры набираются подряд, разделители ставятся сами; ▲▼ шагают по секции, где стоит курсор. Последняя часть — остаток. Если файл короче, записывается одна целая часть. До 10000 частей.
Точки `trim_rows`	trim_rows	-	-	скрыто, когда все из (trim_action = split; trim_split_mode = duration)	Точки - часы, минуты, секунды и миллисекунды: 00:01:02,500. Каждая строка, добавленная кнопкой +, пишется в свой файл.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Плеер — найти точки `trim_player`	mpv_transport	`open`	`open` Открыть в плеере; `both` Взять A и B (скрыто, когда trim_action = split); `mark_in` Отметить IN (скрыто, когда trim_action = split); `mark_out` Отметить OUT (скрыто, когда trim_action = split); `take_a` Взять A (видно, когда все из (trim_action = split; trim_split_mode = points)); `mark_point` Отметить здесь (видно, когда все из (trim_action = split; trim_split_mode = points)); `close` Закрыть плеер	-	Открывает дубль в mpv. Там `l` ставит точку A, второе нажатие — B; кусок между ними играет по кругу, и рез слышно до того, как он сделан. `,` и `.` — шаг в кадр, `b` — воспроизведение назад. Дальше перенести A и B сюда.
Контейнер вывода `trim_container`	выбор	`source`	`source` Как есть; `mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF	скрыто, когда trim_media_kind = audio	Контейнер куска. «Как есть» оставляет контейнер камеры.
Звук `trim_audio`	выбор	`copy`	`copy` Копировать; `none` Убрать	скрыто, когда trim_media_kind = audio	Звук куска: копируется как есть или убирается.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Канал звука `trim_audio_channel`	выбор	`both`	`both` Оба; `first` Первый; `second` Второй; `mono` Свести в моно	видно, когда trim_audio = copy; скрыто, когда trim_media_kind = audio	Какие каналы звука идут в кусок: оба, один из двух или оба, сведённые в моно.
Резка `trim_exact`	выбор	`false`	`true` Точно; `false` Без потерь	видно, когда trim_media_kind = audio	Как режется звуковой файл: ровно по сэмплу, со сжатым звуком, закодированным заново, или без потерь - сжатые пакеты копируются как есть, а точки уходят к границам пакетов.
Формат куска `trim_audio_format`	выбор	`source`	`source` Как есть; `wav` WAV; `flac` FLAC; `alac` ALAC; `m4a` M4A / AAC; `mp3` MP3; `opus` Opus	видно, когда все из (trim_media_kind = audio; trim_exact включено)	Формат, в котором пишется кусок звукового файла. «Как есть» - формат источника, сэмпл в сэмпл, где он без потерь.
MP3 / LAME `trim_mp3_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда все из (trim_media_kind = audio; trim_exact включено; trim_audio_format = mp3)	Качество LAME для куска, записанного в MP3.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Битрейт `trim_audio_bitrate`	выбор	`320k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда trim_audio_format - одно из mp3, opus)	видно, когда все из (trim_media_kind = audio; trim_exact включено; любое из (trim_audio_format - одно из m4a, opus; все из (trim_audio_format = mp3; trim_mp3_preset = bitrate)))	Битрейт куска в AAC, Opus (не выше 256 кбит/с) или MP3 по битрейту (не выше 320 кбит/с).
Телеметрия `trim_data`	флажок	`true`	-	скрыто, когда trim_media_kind = audio	Переносит поток данных в результат — GoPro пишет туда телеметрию, и она иначе исчезает молча. MOV и MP4 такой поток держат, MKV и MXF нет — журнал скажет об этом до запуска. Дорожка таймкода намеренно остаётся: она переезжает значением. Вырезать не проносит поток данных через склейку и говорит об этом.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Субтитры `trim_subtitles`	флажок	`true`	-	скрыто, когда trim_media_kind = audio	Переносит субтитры. У камерных файлов они редки, но DJI пишет субтитрами данные полёта — выбросить их значит потерять журнал полёта. Реплики не копируются, а собираются заново: каждая обрезается по сохраняемому куску и сдвигается на его шкалу времени, так что реплика через IN или OUT сохраняет свою часть внутри, а после «Вырезать» реплики стоят там, куда их ставит склейка. MP4 и MOV пишут их как mov_text, MKV — как SRT; субтитры картинкой остаются, и журнал это скажет.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Таймкод `trim_timecode`	выбор	`shift`	`shift` Сдвинуть; `zero` Обнулить	скрыто, когда trim_media_kind = audio	Таймкод куска: со сдвигом на отрезанное - по нему монтаж синхронизирует камеры - или от нуля.
faststart `trim_faststart`	флажок	`true`	-	скрыто, когда trim_media_kind = audio	Перенести заголовок в начало, чтобы MP4 или MOV начинал играть, не скачавшись целиком.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

ЧАСТОТА - Varispeed и conform в 23,976

Пересчитать FPS - `fps_run`

Прочитать FPS исходника и вывести выбранную частоту.

Вызов: `media_cli.cmd run --operation fps_run`

Режимы ленты: VARISPEED `fps_mode=varispeed`, CONFORM `fps_mode=conform`.

Обёртки: `ff-100to23976-conform`, `ff-100to23976-varispeed`, `ff-11988to23976-conform`, `ff-11988to23976-varispeed`, `ff-17982to23976-conform`, `ff-17982to23976-varispeed`, `ff-24to23976-varispeed`, `ff-25to23976-conform`, `ff-25to23976-varispeed`, `ff-2997to23976-conform`, `ff-2997to23976-varispeed`, `ff-50to23976-conform`, `ff-50to23976-varispeed`, `ff-5994to23976-conform`, `ff-5994to23976-varispeed`.

Пресеты - Профиль вывода `fps_output_profile`

Пресет	Что это	Что ставит
`h264` H.264 MP4	MP4 с H.264 и AAC. Битрейт аудио выбирается ниже.	fps_output_profile=h264, encode_backend=cpu, audio_bitrate=384k, fps_audio_pcm_depth=s24, fps_audio_oversample=x1, crf=14, cq=19, cpu_encoder_preset=medium
`hevc` H.265 MP4	MP4 с H.265/HEVC и AAC. Битрейт аудио выбирается ниже.	fps_output_profile=hevc, encode_backend=cpu, audio_bitrate=384k, fps_audio_pcm_depth=s24, fps_audio_oversample=x1, crf=16, cq=19, cpu_encoder_preset=medium
`prores` ProRes MOV	ProRes 422 в Apple MOV с PCM 16/24/32-float; выбранные 96/192 кГц сохраняются на выходе.	fps_output_profile=prores, fps_audio_pcm_depth=s24, fps_audio_oversample=x2
`prores_mxf` ProRes MXF	ProRes 422 в MXF с PCM 16/24-bit; финальная частота MXF в FFmpeg — 48 кГц.	fps_output_profile=prores_mxf, fps_audio_pcm_depth=s24, fps_audio_oversample=x2
`dnxhr` DNxHR MOV	DNxHR HQX в Apple MOV с PCM 16/24/32-float; выбранные 96/192 кГц сохраняются на выходе.	fps_output_profile=dnxhr, fps_audio_pcm_depth=s24, fps_audio_oversample=x2
`dnxhr_mxf` DNxHR MXF	DNxHR HQX в MXF с PCM 16/24-bit; финальная частота MXF в FFmpeg — 48 кГц.	fps_output_profile=dnxhr_mxf, fps_audio_pcm_depth=s24, fps_audio_oversample=x2

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Целевая частота `fps_target`	выбор	`24000/1001`	`16` 16; `18` 18; `24000/1001` 23.976; `24` 24; `25` 25; `30000/1001` 29.97; `30` 30; `48` 48; `50` 50; `60000/1001` 59.94; `60` 60	-	Частота, с которой файл будет проигрываться. Все кадры остаются: меняется длительность, а не картинка.
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда fps_output_profile - одно из h264, hevc	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).
Кодер `encode_backend`	выбор	`cpu`	`cpu` CPU; `cuda` NVENC; `amd` AMF; `qsv` QSV	видно, когда fps_output_profile - одно из h264, hevc	Выбор backend кодирования. Ниже показываются только родные параметры выбранного backend.
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.
CPU preset `cpu_encoder_preset`	выбор	`medium`	`fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	видно, когда fps_output_profile - одно из h264, hevc; скрыто, когда encode_backend - одно из cuda, qsv, amd	Software x264/x265 preset скорости/сжатия. Более медленные presets тратят больше CPU на более чистое/компактное кодирование при том же CRF.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
NVENC preset `nvenc_preset`	выбор	`p6`	`p5` NVENC p5; `p6` NVENC p6; `p7` NVENC p7	видно, когда fps_output_profile - одно из h264, hevc и encode_backend = cuda	NVIDIA NVENC P-preset. Это шкала аппаратного кодера, не эквивалент CPU slow/slower/veryslow.
QSV preset `qsv_encoder_preset`	выбор	`medium`	`veryfast` veryfast; `faster` faster; `fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	видно, когда fps_output_profile - одно из h264, hevc и encode_backend = qsv	Intel QuickSync preset. У QSV своя FFmpeg-линейка: veryfast, faster, fast, medium, slow, slower, veryslow.
AMF quality `amf_quality`	выбор	`quality`	`speed` speed; `balanced` balanced; `quality` quality	видно, когда fps_output_profile - одно из h264, hevc и encode_backend = amd	AMD AMF quality preset. Управляет AMF режимом speed/balanced/quality; CQ/QP по-прежнему задаёт уровень квантования.
CRF `crf`	число	`14`	-	видно, когда fps_output_profile - одно из h264, hevc; скрыто, когда encode_backend - одно из cuda, qsv, amd	Цель качества для software-кодеков: ниже означает крупнее и чище.
CQ/QP `cq`	число	`14`	-	видно, когда fps_output_profile - одно из h264, hevc и encode_backend - одно из cuda, qsv, amd	Constant Quality для аппаратных кодировщиков.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Разрядность PCM `fps_audio_pcm_depth`	выбор	`s24`	`s16` PCM 16-bit; `s24` PCM 24-bit; `f32` PCM 32-float (доступно, только когда fps_output_profile - одно из prores, dnxhr)	видно, когда fps_output_profile - одно из prores, prores_mxf, dnxhr, dnxhr_mxf	Разрядность звука PCM у ProRes и DNxHR. MXF берёт 16 или 24 бита.
Oversampling аудио `fps_audio_oversample`	выбор	`x2`	`x1` COPY; `x2` 2x 88.2/96; `x4` 4x 176.4/192	видно, когда fps_output_profile - одно из prores, prores_mxf, dnxhr, dnxhr_mxf	Частота внутренней обработки. Финальный звук MXF сводится muxer-ом FFmpeg к 48 кГц.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

ЦВЕТ - LUT, гамма до него, HDR в SDR

Запустить цвет - `grading_run`

Применить выбранную цветовую операцию с текущими overrides.

Вызов: `media_cli.cmd run --operation grading_run`

Обёртки: `ff-grade-lut-only-hevc10`, `ff-grade-lut-only-prores422`, `ff-grade-lut-only-proresLT`, `ff-grade-lut-only-x264`, `ff-grade-preexpose-lut-x264`, `ff-grade-pregamma-lut-hevc10`, `ff-grade-pregamma-lut-prores422`, `ff-grade-pregamma-lut-x264`, `ff-hdr2sdr-hable-hevc10`, `ff-hdr2sdr-hable-x264`.

Пресеты - Цветовой пресет `grading_profile`

Пресет	Что это	Что ставит
`lut_x264` LUT x264	Применить выбранный LUT без pre-gamma и вывести рабочий H.264 MP4.	script_preset=ff-grade-lut-only-x264.cmd, output_container=mp4, pix_fmt=yuv420p, encode_backend=cpu, decode_backend=auto, crf=14, cq=14

Пресет	Что это	Что ставит
`lut_hevc10` LUT HEVC Main10	Применить выбранный LUT без pre-gamma и вывести 10-bit HEVC Main10.	script_preset=ff-grade-lut-only-hevc10.cmd, output_container=mp4, pix_fmt=yuv420p10le, encode_backend=cpu, decode_backend=auto, crf=14, cq=14
`lut_prores422` LUT ProRes 422	Применить выбранный LUT и создать монтажный 10-bit 4:2:2 ProRes intermediate.	script_preset=ff-grade-lut-only-prores422.cmd, output_container=mov, pix_fmt=yuv422p10le, encode_backend=cpu, decode_backend=auto
`preexpose_x264` Pre-expose LUT x264	До LUT применить Gamma 1.15: поднять 18% серый примерно до 22.5%, затем кодировать x264.	script_preset=ff-grade-preexpose-lut-x264.cmd, pre_gamma=1.15, output_container=mp4, pix_fmt=yuv420p, encode_backend=cpu, decode_backend=auto, crf=14, cq=14
`pregamma_hevc10` Pregamma HEVC Main10	До LUT применить Gamma 0.85: опустить 18% серый примерно до 13.3%, затем кодировать HEVC Main10.	script_preset=ff-grade-pregamma-lut-hevc10.cmd, pre_gamma=0.85, output_container=mp4, pix_fmt=yuv420p10le, encode_backend=cpu, decode_backend=auto, crf=14, cq=14
`hdr_hable_x264` HDR to SDR x264	Преобразовать BT.2020/PQ HDR в BT.709 SDR через Hable tonemapping и кодировать x264.	script_preset=ff-hdr2sdr-hable-x264.cmd, output_container=mp4, pix_fmt=yuv420p, encode_backend=cpu, decode_backend=auto, crf=14, cq=14

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
LUT `lut_file`	LUT	-	-	-	По умолчанию LUTs\active.cube или первый LUT-файл из папки LUTs.
Фильтр файлов источника `input_formats`	флажки	-	`mp4` MP4; `mov` MOV; `mkv` MKV; `mxf` MXF; `avi` AVI; `webm` WEBM; `mts` MTS; `m2ts` M2TS	-	Если пусто, команда сама берёт поддерживаемые типы файлов источника.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Gamma до LUT `pre_gamma`	число	-	-	видно, когда script_preset - одно из ff-grade-preexpose-lut-x264.cmd, ff-grade-pregamma-lut-x264.cmd, ff-grade-pregamma-lut-hevc10.cmd, ff-grade-pregamma-lut-prores422.cmd	Gamma применяется до LUT. Значения выше 1 поднимают 18% серый, ниже 1 — опускают. Preview показывает технический ориентир до LUT.
Цели цветового вывода `target_codecs`	флажки	-	`x264` x264 / H.264; `hevc_x265` x265 / HEVC; `svt_av1` AV1; `prores_422` ProRes 422; `prores_lt` ProRes LT	-	Один или несколько выходных таргетов. Каждый таргет пишется отдельно для каждого файла Источника.
Кодер `encode_backend`	выбор	`cpu`	`cpu` CPU; `cuda` NVENC; `amd` AMF; `qsv` QSV	-	Выбор backend кодирования. Ниже показываются только родные параметры выбранного backend.
Декодировать через `decode_backend`	выбор	`auto`	`auto` CPU/Auto; `cuda` CUDA; `qsv` QuickSync; `amd` AMD/D3D11VA; `dav1d` dav1d AV1	-	Предпочтение аппаратного или программного декодирования перед кодированием.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Аудио `audio_mode`	выбор	`source`	`source` COPY (недоступно, когда output_container = mxf); `aac_320` AAC override (недоступно, когда output_container = mxf); `mp3` MP3 (недоступно, когда output_container = mxf); `flac` FLAC (недоступно, когда output_container - одно из mxf, mov); `pcm_s16` PCM 16-bit; `pcm_s24` PCM 24-bit; `pcm_f32` PCM float (недоступно, когда output_container = mxf); `none` Без аудио	-	Политика аудио результата: copy потока, lossy/lossless-кодирование, разрядность PCM или без аудио. Для MXF доступны только совместимые PCM-режимы.
Частота дискретизации `audio_sample_rate`	выбор	`source`	`source` COPY; `44100` 44.1 kHz; `48000` 48 kHz; `88200` 88.2 kHz; `96000` 96 kHz; `176400` 176.4 kHz; `192000` 192 kHz	видно, когда audio_mode - одно из aac_320, mp3, flac, pcm_s16, pcm_s24, pcm_f32	Целевая частота дискретизации. Родная оставляет исходную частоту.
MP3 / LAME `mp3_lame_preset`	выбор	`v0`	`v0` VBR V0; `insane` Insane 320; `bitrate` По битрейту	видно, когда audio_mode = mp3	Режим качества LAME для MP3.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Битрейт аудио `audio_bitrate`	выбор	`384k`	`192k` 192 kbps; `256k` 256 kbps; `320k` 320 kbps; `384k` 384 kbps (недоступно, когда любое из (audio_mode = mp3; audio_format = mp3))	видно, когда любое из (audio_mode = aac_320; все из (audio_mode = mp3; mp3_lame_preset = bitrate))	Битрейт сжатого звука: AAC, Opus (не выше 256 кбит/с) и MP3, когда его пресет LAME - «По битрейту» (не выше 320 кбит/с).
Контейнер `output_container`	выбор	`mp4`	`mp4` MP4; `mkv` MKV; `mov` MOV	-	Контейнер результата, если выбранное семейство кодеков его допускает.
CPU preset `cpu_encoder_preset`	выбор	`medium`	`fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	скрыто, когда encode_backend - одно из cuda, qsv, amd	Software x264/x265 preset скорости/сжатия. Более медленные presets тратят больше CPU на более чистое/компактное кодирование при том же CRF.
NVENC preset `nvenc_preset`	выбор	`p6`	`p5` NVENC p5; `p6` NVENC p6; `p7` NVENC p7	видно, когда encode_backend = cuda	NVIDIA NVENC P-preset. Это шкала аппаратного кодера, не эквивалент CPU slow/slower/veryslow.
QSV preset `qsv_encoder_preset`	выбор	`medium`	`veryfast` veryfast; `faster` faster; `fast` fast; `medium` medium; `slow` slow; `slower` slower; `veryslow` veryslow	видно, когда encode_backend = qsv	Intel QuickSync preset. У QSV своя FFmpeg-линейка: veryfast, faster, fast, medium, slow, slower, veryslow.
AMF quality `amf_quality`	выбор	`quality`	`speed` speed; `balanced` balanced; `quality` quality	видно, когда encode_backend = amd	AMD AMF quality preset. Управляет AMF режимом speed/balanced/quality; CQ/QP по-прежнему задаёт уровень квантования.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Разрешение `output_resolution`	выбор	`source`	`source` Исходное; `2160` 2160p; `1440` 1440p; `1080` 1080p; `720` 720p; `480` 480p	-	Необязательное масштабирование. Исходное сохраняет размер кадра.
Чересстрочный источник (редкое) `scan_mode`	выбор	`double`	`keep` Оставить поля; `progressive` Прогрессив; `double` Прогрессив x2	-	Что станет с источником из двух полей (25i, 29,97i, 50i). «Прогрессив x2», по умолчанию: bwdif делает кадр из каждого поля, частота вдвое выше, движение целиком - 50i становится 50p. «Прогрессив»: цельный кадр из каждого кадра, частота прежняя. «Оставить поля»: поля и их порядок сохраняются - для чересстрочного эфира. Прогрессивный источник не трогается.
SVT-AV1 preset `av1_preset`	выбор	`6`	`4` 4 quality; `6` 6 balanced; `8` 8 fast	видно, когда target_codecs - одно из svt_av1; скрыто, когда encode_backend - одно из cuda, qsv, amd	Preset скорости/качества software SVT-AV1. Аппаратные AV1-кодеры используют свои шкалы NVENC/QSV/AMF.
Pixel format `pix_fmt`	выбор	`auto`	`auto` auto; `yuv420p` yuv420p; `yuv420p10le` yuv420p10le; `yuv422p` yuv422p; `yuv422p10le` yuv422p10le	-	Pixel format и разрядность результата. Значение должно соответствовать выбранному кодеку и монтажному либо delivery-сценарию.
CRF `crf`	число	`14`	-	скрыто, когда encode_backend - одно из cuda, qsv, amd	Цель качества для software-кодеков: ниже означает крупнее и чище.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
CQ/QP `cq`	число	`14`	-	видно, когда encode_backend - одно из cuda, qsv, amd	Constant Quality для аппаратных кодировщиков.
Перезаписывать `overwrite`	флажок	`false`	-	-	Разрешить FFmpeg заменять существующие выходные файлы.
Тест: первый файл `limit_first_file`	флажок	`false`	-	-	Обработать только первый найденный файл, удобно для проверки пресета перед всей папкой.
Только показать команду `dry_run`	флажок	`false`	-	-	Показать команды без обработки медиа.

СКАЧАТЬ - Видео или звук по ссылке

Скачать - `youtube_run`

Запустить выбранный профиль yt-dlp.

Вызов: `media_cli.cmd run --operation youtube_run`

Режимы ленты: ЛУЧШЕЕ MP4 `youtube_profile=best_mp4`, НАДЁЖНОЕ MP4 `youtube_profile=safe_mp4`, СВОЁ ВИДЕО `youtube_profile=video_custom`, ДЛЯ ЗАГРУЗКИ `youtube_profile=upload_ready`, СВОЁ АУДИО `youtube_profile=audio_custom`, СУБТИТРЫ `youtube_profile=subs_only`.

Обёртки: `yt-1080p-avc-mp4`, `yt-480p-aac`, `yt-batch-480p-aac`, `yt-batch-best-mp4`, `yt-best-av1`, `yt-best-m4a`, `yt-best-mp4`, `yt-best-mp4-safari`, `yt-best-mp4-safe`, `yt-clip-sections`, `yt-embed-subs`, `yt-mp3`, `yt-opus`, `yt-playlist-1080p-avc`, `yt-playlist-best`, `yt-subs-only`, `yt-webm-vp9`.

Пресеты - Профиль скачивания `youtube_preset`

Пресет	Что это	Что ставит
`best_video` Best MP4 video	Лучшее доступное MP4/M4A для обычной загрузки.	youtube_profile=best_mp4, youtube_source_mode=single, youtube_resolution=best, youtube_video_codec=auto, youtube_audio_format=source, youtube_container=mp4, youtube_client=default, youtube_options=safe_retries,embed_metadata,no_playlist, subtitle_options=, parallel_fragments=8, download_archive=false

Пресет	Что это	Что ставит
`upload_1080_avc` 1080p AVC MP4	Предпочтение 1080p AVC для совместимости и заливки.	youtube_profile=upload_ready, youtube_source_mode=single, youtube_resolution=1080, youtube_video_codec=avc, youtube_audio_format=source, youtube_container=mp4, youtube_options=safe_retries,embed_metadata,no_playlist, subtitle_options=, parallel_fragments=8, download_archive=false
`safe_safari` Safe Safari	Медленный Safari-клиент, повторы и IPv4 для капризных ссылок.	youtube_profile=safe_mp4, youtube_source_mode=single, youtube_resolution=best, youtube_video_codec=auto, youtube_audio_format=source, youtube_container=mp4, youtube_client=safari, youtube_options=safe_retries,force_ipv4,embed_metadata,no_playlist, subtitle_options=, parallel_fragments=1, download_archive=false
`av1_if_available` AV1 если доступен	Предпочитать AV1-видео, оставляя MP4-контейнер.	youtube_profile=video_custom, youtube_source_mode=single, youtube_resolution=best, youtube_video_codec=av1, youtube_audio_format=source, youtube_container=mp4, youtube_options=safe_retries,embed_metadata,no_playlist, subtitle_options=, parallel_fragments=8, download_archive=false
`audio_mp3` Audio MP3	Извлечение только аудио в MP3.	youtube_profile=audio_custom, youtube_source_mode=single, youtube_audio_format=mp3, youtube_options=safe_retries,embed_metadata,no_playlist, subtitle_options=, parallel_fragments=8, download_archive=false
`batch_best` Batch из urls.txt	Читать ссылки из input\urls.txt.	youtube_profile=best_mp4, youtube_source_mode=batch, youtube_resolution=best, youtube_video_codec=auto, youtube_audio_format=source, youtube_container=mp4, youtube_options=safe_retries,embed_metadata, subtitle_options=, parallel_fragments=8, download_archive=false

Поля

Поле	Вид	По умолчанию	Варианты	Когда	Что это
URL `youtube_url`	текст	-	-	-	Ссылка на видео, а при Источнике «Плейлист» - на плейлист. yt-dlp берёт YouTube и большинство видеосайтов.
Фрагмент (редкое) `youtube_sections`	текст	-	-	-	Только часть видео, в записи yt-dlp: *начало-конец. Пусто - видео целиком.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Диапазон списка (редкое) `youtube_playlist_items`	текст	-	-	видно, когда youtube_source_mode = playlist	Какие видео плейлиста, по их номерам в нём: 1-10,15 или -5: - последние пять. Пусто - плейлист целиком.
Файл ссылок `batch_file`	файл	`input\urls.txt`	-	-	Текстовый файл ссылок, по одной в строке. По умолчанию input\urls.txt - это urls.txt Источника, где бы Источник ни был.
Разрешение `youtube_resolution`	выбор	`best`	`best` Лучшее; `2160` 2160p; `1440` 1440p; `1080` 1080p; `720` 720p; `480` 480p	-	Предел разрешения: берётся лучшее не выше него. «Лучшее» - самое высокое, какое есть у сайта.
Видео кодек `youtube_video_codec`	выбор	`auto`	`auto` Auto; `avc` AVC / H.264; `av1` AV1; `vp9` VP9	-	Кодек, которому отдаётся предпочтение, а не требование: без него придёт лучший другой. H.264 в MP4 приходит со звуком AAC - для монтажа и форм загрузки, которые не берут Opus.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Формат аудио `youtube_audio_format`	выбор	`source`	`source` Родной; `m4a` M4A; `opus` Opus; `mp3` MP3	-	Звук «Своего аудио». «Родной» - лучший звук, как его хранит сайт, без перекодирования. M4A - AAC сайта, если он есть, иначе перекодирование. Opus и MP3 (VBR V0) - звук в этом кодеке, FFmpeg перекодирует, если у сайта другой.
Контейнер `youtube_container`	выбор	`mp4`	`mp4` MP4; `mkv` MKV; `webm` WEBM	-	Контейнер, в который сводятся картинка и звук.
Источник `youtube_source_mode`	выбор	`single`	`single` Одна ссылка; `batch` input\urls.txt; `playlist` Плейлист	-	Откуда ссылки: одна ссылка, список input\urls.txt или весь плейлист.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Тюнинг скачивания `youtube_options`	флажки	-	`safe_retries` Повторы; `force_ipv4` IPv4; `embed_metadata` Метаданные + обложка; `no_playlist` Без плейлиста	-	«Повторы» - десять попыток на запрос и на фрагмент, через две секунды. IPv4 - только соединения IPv4. «Метаданные + обложка» - теги и обложка в файл. «Без плейлиста» - для ссылок из input\urls.txt: из ссылки на видео в плейлисте только это видео; одна ссылка всегда берёт только своё видео, «Плейлист» - весь список.
Субтитры `subtitle_options`	флажки	-	`write_subs` Ручные; `write_auto_subs` Авто; `embed_subs` Встроить	-	«Ручные» - субтитры, написанные людьми, файлами рядом с видео. «Авто» - сделанные самим сайтом. «Встроить» - субтитры внутрь файла видео.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Потоки `parallel_fragments`	число	`8`	-	-	Сколько фрагментов одного видео качается одновременно (yt-dlp -N). Больше - быстрее на хорошем канале; сайт с ограничениями может ответить ошибками.
Клиент (редкое) `youtube_client`	выбор	`default`	`default` Default; `safari` Safari; `android` Android	-	Каким плеером YouTube представляется yt-dlp. Default подходит большинству ссылок; Safari или Android - когда не хватает форматов или режется скорость.
Языки субтитров (редкое) `subtitle_languages`	текст	`ru,en.*`	-	-	Языки субтитров через запятую; можно шаблоном: en.* берёт en, en-US, en-GB.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Cookies (редкое) `cookies_source`	выбор	`file`	`none` Нет; `chrome` Chrome; `edge` Edge; `firefox` Firefox; `brave` Brave; `file` Файл	-	Откуда yt-dlp берёт cookies для ссылки, которой нужен вход. Браузер: yt-dlp читает его профиль в момент загрузки, программа ничего не хранит; Chrome и Edge закрывают cookies своим шифрованием, и yt-dlp может их не прочитать - надёжнее Firefox. Файл: копия, взятая кнопкой «Взять под DPAPI», хранится запечатанной для этой учётной записи Windows; без неё - файл поля «Файл cookies», если он есть. Нет: без cookies.
Файл cookies (редкое) `cookies_path`	файл	`config\cookies.txt`	-	-	Файл cookies браузера (формат Netscape, cookies.txt) для ссылки, которой нужен вход. «Взять под DPAPI» хранит его запечатанную копию, и открытый файл больше не нужен; без копии файл читается как есть, если он есть.

Поле	Вид	По умолчанию	Варианты	Когда	Что это
Вести архив скачиваний (редкое) `download_archive`	флажок	`false`	-	-	Запоминает скачанное в config\download-archive.txt; ссылка, которая там уже есть, пропускается.
Только показать команду `youtube_dry_run`	флажок	`false`	-	-	Показывает команду yt-dlp и ничего не запускает.

СВЕДЕНИЯ

Проверить источник - `probe_source`

Один раз запустить ffprobe по Источнику, вывести таблицу и обновить строку параметров.

Вызов: `media_cli.cmd run --operation probe_source`

Инфо об источниках - `source_info`

Прогнать ffprobe по Источнику и вывести все аудиопотоки: codec, sample_fmt, sample rate, bits, channels, bitrate, language/title.

Вызов: `media_cli.cmd run --operation source_info`

СЕРВИС: проверки и установка

Инвентаризация - `inventory`

Посчитать файлы в Источнике, Назначении и LUTs.

Вызов: `media_cli.cmd run --operation inventory`

Самопроверка - `selftest`

Проверить ffmpeg, ffprobe, yt-dlp и Deno.

Вызов: `media_cli.cmd run --operation selftest`

Обёртки: `ff-selftest`, `yt-selftest`.

Проверка профилей - `profile_doctor`

Проверить операции GUI, script wrappers и общий каталог пресетов.

Вызов: `media_cli.cmd run --operation profile_doctor`

Возможности оборудования - `hardware_capabilities`

Короткий FFmpeg smoke для CPU, QSV/NVENC/AMF H.264 и HEVC, SVT-AV1 и dav1d.

Вызов: `media_cli.cmd run --operation hardware_capabilities`

Тест матрицы пресетов - `preset_test_matrix`

Прогнать portable smoke на 1-секундных файлах по audio, remux, FPS, LUT, архиву, монтажу, хранению и публикации.

Вызов: `media_cli.cmd run --operation preset_test_matrix`

Установить 7-Zip - `install_7zip`

Установить или обновить portable 7-Zip; нужен для FFmpeg 7z payloads.

Вызов: `media_cli.cmd run --operation install_7zip`

Установить FFmpeg - `install_ffmpeg`

Установить или обновить переносную FFmpeg от Gyan (GPL) в Tools\ffmpeg по драйверу NVIDIA: самая новая стабильная, не ниже 9.0.2, без карты NVIDIA и с LTS-драйвером 610.88 и новее; ниже него 8.0.1 (570-610.87) или 7.1, чтобы работал NVENC, и доступны не все операции.

Вызов: `media_cli.cmd run --operation install_ffmpeg`

Установить yt-dlp - `install_ytdlp`

Установить или обновить portable yt-dlp и Deno в Tools.

Вызов: `media_cli.cmd run --operation install_ytdlp`

Обёртки: `yt-update`.

Установить mpv - `install_mpv`

Установить или обновить portable mpv в Tools/mpv: плеер нарезки.

Вызов: `media_cli.cmd run --operation install_mpv`

Установить проявку ProRes RAW - `install_cineraw`

Собрать cineraw из libcraw в Tools/cineraw из исходников: переносной MSYS2 для сборки (около 185 МБ загрузки, 5 минут), удаляется после; cineraw проявляет ProRes RAW на видеокарте с Vulkan 1.4 (NVIDIA RTX).

Вызов: `media_cli.cmd run --operation install_cineraw`

Кнопки у полей

Форматы ссылки - `youtube_formats`

Какие форматы есть у ссылки - разрешение, кодеки, размер - списком в журнале, ничего не скачивается. Взять один из них - профилем «Своё видео».

Рядом с полем: `youtube_url`. Вызов: `media_cli.cmd run --operation youtube_formats`.

Взять под DPAPI - `youtube_cookies_import`

Файл поля «Файл cookies» запечатывается DPAPI для этой учётной записи Windows в config\cookies.dpapi: загрузки с Cookies - «Файл» берут эту копию, и открытый файл больше не нужен.

Рядом с полем: `cookies_path`. Вызов: `media_cli.cmd run --operation youtube_cookies_import`.

Что окно делает само

Рабочие папки окно очищает само, через Корзину; консольная команда эти операции не запускает - они удаляют насовсем.

- Очистить Источник - `cleanup_source`
- Очистить Назначение - `cleanup_transcoded`
- Очистить GUI workspace - `cleanup_workspace`