



ARRI ALEXA 35 ARRIRAW HDE Transcoder 1.6.1

Quick Guide

Date: 4 May 2024

1 About

All recording formats of the ALEXA 35 are now stored in MXF containers. The same container is used when ARRIRAW clips are encoded with CODEX High Density Encoding (HDE). HDE is a lossless, variable bitrate encoding scheme, optimized for Bayer pattern (ARRIRAW) images. The amount of data reduction varies between 40 and 50%, depending on the image content. Once decoded, you get a bit-exact match to the original uncompressed ARRIRAW images.

In 2024, J. Eggleton and D. Holroyd, the CODEX engineers behind HDE, received a Technical Achievement Award from the Academy of Motion Picture Arts and Sciences. It's that good.

2 HDE Transcoding Options

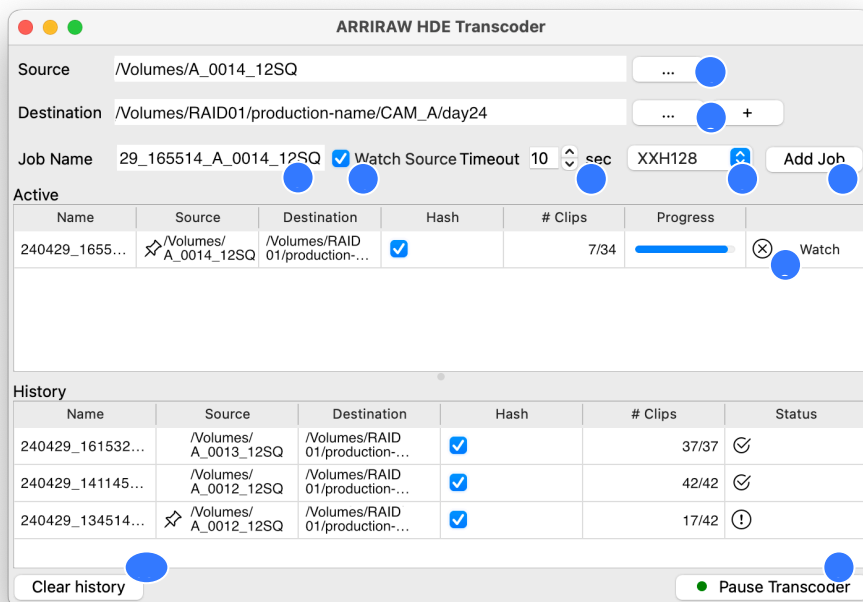
The camera does not record HDE files to the recording media. It records ARRIRAW clips, which then have to be converted to HDE. To get from ALEXA 35 MXF/ARRIRAW to MXF/HDE, you can use:

- The CODEX [Device Manager](#), a software for macOS, offers on the fly HDE encoding through a virtual file system. The files need to be copied from the recording media with data wrangling software (Silverstack, Offshoot, ShotPut Pro, YoYotta,... see check the [CODEX website](#) for more details).
- The [ARRIRAW HDE Transcoder](#), our simple converter tool which uses the original CODEX HDE engine.

2.1 ARRIRAW HDE Transcoder – Graphical User Interface

The ARRIRAW HDE Transcoder with GUI is available for macOS and Windows. It can read ALEXA 35 ARRIRAW clips directly from the camera recording media or any other folder. The software allows an output to multiple destinations. For downstream checksum verification, it can generate original checksums in different hash types (SHA1, MD5, xxh64 etc) and media hash list versions (mhl or ascmhl).

Setting up a Transcode Job



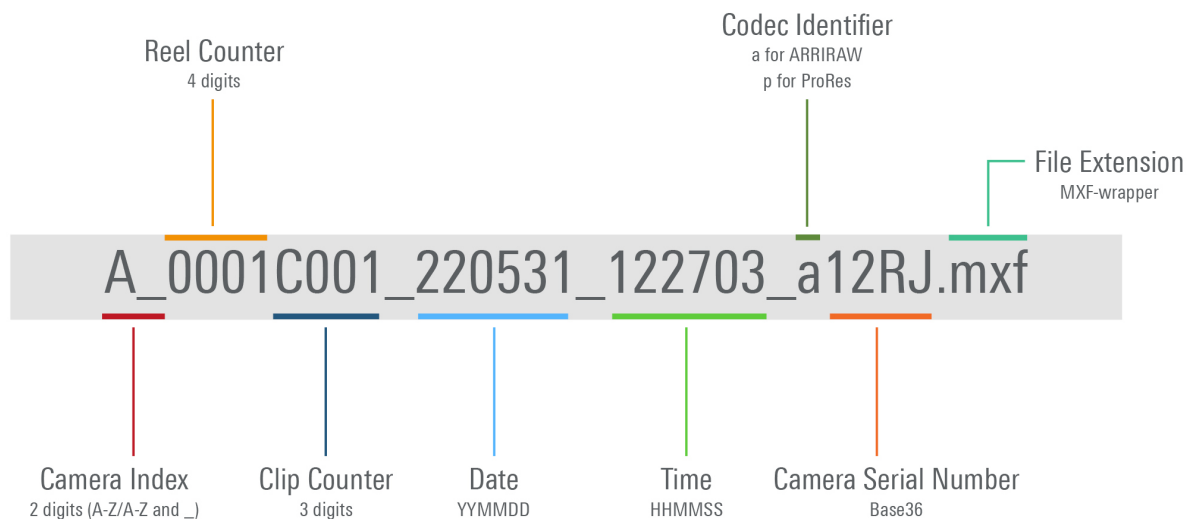
- (1) Select a 'Source' folder which may contain clips or a folder containing more than one reel of clips. All other files (MXF/ProRes, ale, bin, and non-camera data) will be copied to the destination.
- (2) Select a 'Destination' folder where the output will be written. The software will mirror the structure under the source directory. Add/remove destinations with '+' or '-'.
- (3) The 'Job Name' is automatically created based on the source folder name and can be changed.

- (4) The 'Watch Source' option tells the Transcoder to encode all files in a folder and then keep monitoring for new files. A 'watch' job is identified with a pin in the 'Active' job list.
- (5) Use the Timeout setting in case the application produces a 'not enough data' error message. This may be the case the source cannot be accessed, or the data rates drops significantly.
- (6) Use the dropdown menu next to the Timeout box to select from the available hash types and media hash list types (MHL or ASCMHL).
- (7) Use 'Add Job' to start transcoding or add a job to the queue.
- (8) The 'X' icon in the 'Active' queue to abort a job or stop an active 'Watch Source' job.
- (9) Use 'Pause Transcoder' to temporarily pause an active job. The encoder will finish the current clip and then wait until you click the button again to 'Resume'.
- (10) 'Clear History' will clear the jobs shown in the job 'History' list.

To access the ARRIRAW HDE Transcoder log files, use 'File > Show Log Files...' from the menu bar.

File name and Clip Name

MXF files from an ALEXA 35 can contain uncompressed ARRIRAW data or Apple ProRes. To allow differentiation without opening the files, we offer a codec identifier in the file name:



H for HDE

When an ARRIRAW clip is converted to HDE, we change the codec identifier to 'h'. This also applies to the "name" and "file name" metadata fields in the MXF file and the entries in the .ALE file, as editing software typically will use the metadata or the ale file to find the files.

The Transcoder cannot detect the codec identifier if the file name pattern was modified. In that case, the software will append '_hde' to the file name. You can also force the use of the '_hde' suffix by activating 'File > Output File Name:...' in the menu bar. This, however, is a legacy setting and we do not recommend to use it unless you are absolutely sure this is what you need.

2.2 ARRIRAW HDE Transcoder – Command Line Version

The ARRIRAW HDE Transcoder as CMD version is available for macOS, Windows, and Linux.

```
arrirawhde [options] -i <input.mxf | input_folder | job.json> -o <output.mxf | output_folder>
```

options:

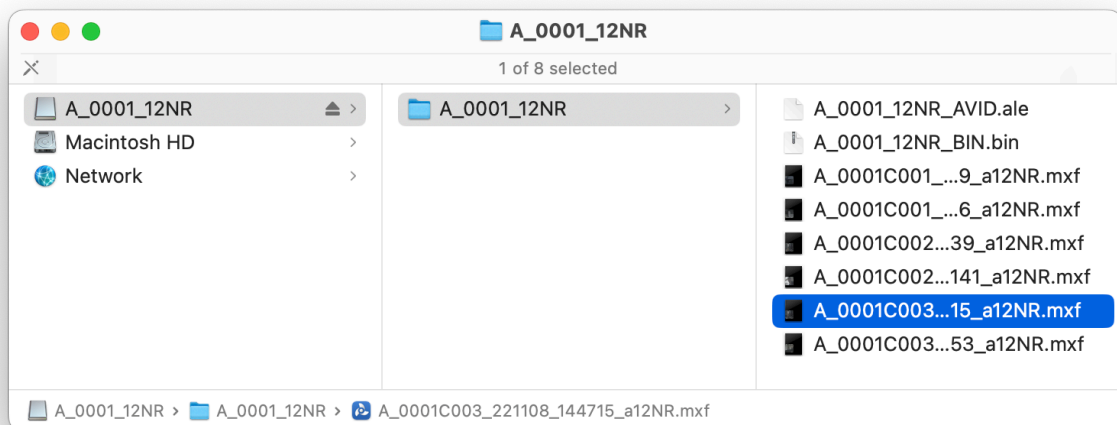
- a activate watch folder
- c <value>-<type> sets the media hash list and checksum type
m1 for mhl v1, m2 for ascmhl v2. md5, sha1, sha256, c4, xxh64, xxh3, xxh128
Example: -c m2-xxh64
- r <path/file> create a json report with specified file name at the specified path
- t <value > demux timeout in seconds
- l legacy output file name. Appends _hde instead of using the hde identifier _h####.mxf.
- v verbose mode
- eula print end user license agreement

In addition to the functions of the GUI version, the CMD version allows you to specify a single clip or a job file for batch processing. Appending another -o output.mxf/folder to the command line will write the HDE output to an additional destination.

Command Line Examples

Below are some examples for use of the command line tool. We'll use a Compact Drive named A_0001_12NR and two shuttle disks called TD1 and TD2, which are mounted at /Volumes/A_0001_12NR/, /Volumes/TD1/ and /Volumes/TD2/.

Remember that the camera assigns the reel name to the drive and a folder on the root level, which then contains the recorded data!



Note: In the examples on the next pages, we use a path notation for macOS or Linux. For Windows, you'd need to specify a path like X:\A_0001_12NR\A_0001_12NR\A_0001C001_240504_104800_a12NR.mxf.

Encoding a Single Clip (File)

```
arrirawhde -i /Volumes/A_0001_12NR/A_0001_12NR/A_0001C001_240504_104800_a12NR.mxf -o /Volumes/TD1/
```

This will convert clip A_0001C001_240504_104800_a12NR.mxf in folder A_0001_12NR on drive A_0001_12NR and output an HDE clip named A_0001C001_240504_104800_h12NR.mxf on the root level of shuttle disk TD1. The ALE file also will be copied to the destination.

Encoding a Reel (Folder)

```
arrirawhde -v -c m2-xxh64 -i /Volumes/A_0001_12NR/ -o /Volumes/TD1/ -o /Volumes/TD2/
```

This will read all contents on drive A_0001_12NR and mirror the folder structure on destination drives TD1 and TD2. ARRIRAW files will be encoded to HDE. Other files will be copied. Verbose mode will show information on every clip that is encoded. Next to that, the software will create an ascmhl folder including a chain file (mhl history) and an ascmhl containing xxHash64 hash values for all transferred files.

Encoding Data Specified in a Job File

```
arrirawhde -v -i /Users/Demo/day001-transcode.json
```

This will convert the data listed in the job file named day001-transcode.json. For this example, we'll assume that all camera reels shot on day 1 have been offloaded to an on set RAID mounted at /Volumes/OSRAID/ into folders day001/A-CAM, B-CAM etc. Here's what the json file could look like:

```
{
  "name": "The Example Job - Day1 A-CAM all and B-CAM selected test shot only",
  "checksum": "m2-xxh64",
  "report": "/Volumes/TD1/transcoder-reports/day001-report.json",
  "items":
  [
    {
      "source": "/Volumes/OSRAID/day001/A-CAM/",
      "target_dir": "/Volumes/TD1/render-out/day001/A-CAM/"
    },
    {
      "source": "/Volumes/OSRAID/day001/B-CAM/B_0001_12SQ/B_0001C003_240408_140120_a12SQ.mxf",
      "target_dir": "/Volumes/TD1/render-out/day001/B-CAM/B_0001_12SQ/"
    }
  ]
}
```

3 Checksums

The transcoder process reads an MXF/ARRIRAW file, encodes the image essence writes a new MXF/HDE file. Since the two files are inherently different, it's not possible to run a source/destination verification as you'd do if you copied ARRIRAW from camera media to a backup drive. The same applies to the ALE file, which is also updated so it links to the new HDE files.

It still is beneficial to create checksums and store them in an ascmhl (or mhl) file. The checksum output by the transcode process is your new master file checksum. This allows you to identify file transfer errors as the unmodified data is handed on. It should even allow you to uncover I/O issues that occurred when the HDE file are initially written.

4 Known Issues

ASCMHL Chain Broken

If the ARRIRAW HDE Transcoder reads from a source that contains an ASC Media Hash List, none of the clip checksums nor the ALE file can be verified. As these files are intentionally changed by the transcoder, it can only create new original checksums and break the history.

Encoding the same ARRIRAW clip twice results in different checksums

The MXF specification requires that files with the same essence, but different creation and/or modification time have a different unique identifier. If the same file is encoded at a different time, the resulting checksum therefore should be different. If the transcoder writes HDE files to more than one destination in the same job, the will be created simultaneously and you will see matching checksums.

HDE CRC image checksum

HDE output files include a secondary image checksum to allow for checksum validation of losslessly compressed ARRIRAW pictures without the need to invert the lossless encoding. The definition will be added to SMPTE RDD 55. Please contact digitalworkflow@arri.de if you create software and need assistance to access this information.

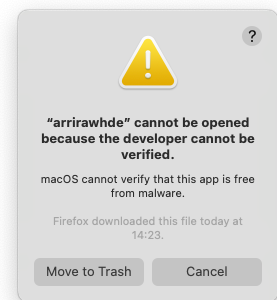
Legacy _hde names not linked in ALE File

The ALE file is always patched to reference clip names and file names using the _h codec identifier, even if the application is set to use the legacy _hde suffix.

macOS and First Run of the Command Line Version

The first time you try to run the command line version on macOS, you will get a popup that the software cannot be opened because the developer cannot be verified. Click 'Cancel', then open the System Preferences, go to 'Privacy and Security', scroll down to Security where, you should see information that "arrirawhde" was blocked from use... Please click 'Allow Anyway'. Next time you start the software, you have to acknowledge that you want to open it once more in a popup message.

This process will repeat itself when you upgrade to a new version of the ARRIRAW HDE Transcoder.



macOS and Installation of the GUI Version

If the ARRIRAW HDE Transcoder installation fails on an Apple computer that was updated to macOS Ventura, please check that Installer.app has Full Disk Access in System Prefs > Security & Privacy and launch the installer .PKG file from within the applications folder.

macOS and System Sleep

We've received reports that a Mac will enter sleep mode if the screen is locked or left unattended for an extended period. We currently don't have a solution for that other than using alternative methods to lock the screen (cf. babyproof on the app store).

macOS and Spaces

If you are using 'Spaces' on the Mac, the arrirawhde process will be moved to the efficiency cores once you move the focus to another Space. To avoid this behaviour, don't switch Spaces or right click on the ARRIRAW HDE Transcoder dock icon and assign the app to all Spaces.

mhl tool version

The mhl file does not show the correct software arrirawhde software version information (1.0.0 instead of 1.6.1). This will be fixed in the next release.

Processing order

Especially for jobs using the 'Watch Source' option, the files may not be processed in the expected order.

Temporary Files and Watch Source

The ARRIRAW HDE Transcoder lacks the function to ignore and exclude certain files/filetypes. This is an issue when you run a 'Watch Source' job and the "copy tool" that transfers files into the source folder uses temporary files (e.g. .pfncopy files created by Pomfort). If the transcoder happens to come across such a temporary file, it's going to detect it as foreign file and copy it to the destination. While it's doing that, the temporary file may not be changed by the copy tool. We're working on resolving this in an upcoming version.

5 Change Notes

Version 1.6.1

- Changed the mhl file name to <numbering>_<source-folder-name>_<YYMMDD_hhmmss>.mhl
- Legacy support: Reduced the hash type selection for MHL v1 to prevent verification errors resulting from unsupported hash types.
- Legacy support: Renamed the MHL hash type ID for "xxh64" to "xxhash64be" so MHL v1 tools will no longer fail to verify.
- Bugs fixed: "illegal instruction" and "segmentation fault" under Linux.
- Improvement: The CentOS Linux version now should run on RHEL and Rocky Linux as well.
- Improvement: The job table now shows an info for multi-destination jobs.
- Other minor fixes and improvements.

Version 1.6

- Bug fixed: wrong bytes 13,14,15 calculation for material and source UMIDs.
- Improved GUI layout under MacOS.
- Improved error handling if a destination does not have enough disk space.
- Added a warning if a multi-destination setup points to the same location more than once.

Version 1.5

- Multiple destinations feature added.
- Bug fixed: the warning dialog does not keep a resized state.
- Disable a filename suffix changing, ALE patching, clip metadata patching.
- Bug fixed: truncated output for SHA1 hash.
- Bug fixed: incorrect C4 calculation for big files.
- Skip files with name started with '.'
- C4 hashing now uses always for mhl chain items.

Version 1.4

- Output files now include an HDE CRC image checksum (to be added to the RDD)
- The application now copies and hashes all files from the source path.
- The _h encoding identifier is now carried forward into the clip name metadata and the ALE file.
- New hashers added: SHA1, SHA256, C4, XXH3, XXH128.
- Added maintenance for ascmhl_chain.xml file.
- Added ascmhl file sequencing.
- Handling for Ctrl-C and other signals has been added to console application.
- json report for command line watchfolder mode now updated after each file.
- Work in progress: Validation, if the source path contains an ascmhl hashlist.
- The behavior of the processed clip counter has been changed.

Version 1.3 beta

- The "clear log" button caption changed to "clear history".
- Keep history unless it is cleared by the user implemented.
- Spaces into the ihashlistî tag of ascmhl have removed.
- Menu: File > Show Log File(s) replacement.
- Output file name correction to use _h encoding identifier by default instead of _hde suffix.
- Imprint tool's build number into the mhl/ascmhl.
- Mhl v1 support added.
- Checksums for .ale and .bin files calculated now.
- Preserve Clip (UU)ID in Material Package UID.
- Bug fixed: "ascmhl conformity - leading zeros of xxhash64 checksums missing".

6 References

ASC Media Hash List: <https://theasc.com/asc/asc-media-hash-list> and <https://github.com/ascmhc/mhl>

SMPTE Registered Disclosure Documents: <https://www.smpte.org/standards/document-index/rdd>

RDD 54 - Mapping ARRIRAW Bitstreams into the MXF Generic Container

RDD 55 - MXF Carriage of ARRI Camera System Metadata

RDD 51 - High Density Image Encoding for ARRIRAW Files

For a list of tools supporting ALEXA 35 recording formats, HDE, and look files, please go to [ARRI.com > Learn & Help > Camera & Workflow > ALEXA 35 Workflows](#)

To learn more about HDE or to register for the new Device Manager public beta please contact support@codex.online!

For questions and feedback on the ARRIRAW HDE Transcoder, please send us a message to digitalworkflow@arri.de!