SMPTE REGISTERED DISCLOSURE DOCUMENT

SMPTE RDD 30:2014

ARRIRAW Image File Structure and Interpretation Supporting Deferred Demosaicing to a Logarithmic Encoding

The attached document is a Registered Disclosure Document prepared by the proponent identified below. It has been examined by the appropriate SMPTE Technology Committee and is believed to contain adequate information to satisfy the objectives defined in the Scope, and to be technically consistent.

This document is NOT a Standard, Recommended Practice or Engineering Guideline, and does NOT imply a finding or representation of the Society.

Errors in this document should be reported to the proponent identified below, with a copy to eng@smpte.org.

This document is intended to support the development of applications that read and process ARRIRAW image files. It is not intended to support the development of hardware or software applications that create ARRIRAW image files, and creation of such files is reserved to individuals and organizations that have entered into agreements with the proponent identified below for such file creation. All inquiries in respect of this document (other than reporting of errors in the document, which should be handled as described in the prior paragraph), including inquiries as to intellectual property requirements that may be attached to use of the disclosed technology, should be addressed to the proponent identified below.

Proponent contact information:

Joseph Goldstone
Arnold & Richter Cine Technik GmbH & Co. Betriebs (KG)
Turkenstraße 89
D-80799 München
Germany

Email: jgoldstone@arri.com

Approved
August 16, 2014

Table of Contents

Page

Introduction 3
1 Scope 4
2 Normative References 4
3 Terms and Definitions 4
4 Acronyms and Abbreviations 4
5 Trademarks 6
6 Definition of ARRIRAW Image File Structure, Image Metadata and Image Data 6
6.1 ARRIRAW Image Metadata Header Structure 6
6.2 ARRIRAW Image Data 21
Annex A Bibliography (Informative) 25

Introduction

This document defines ARRIRAW image file structure and an associated image data and metadata interpretation that allows storage of photomosaic data produced by ARRI cameras, along with algorithms for processing that stored photomosaic data to produce scene-referred logarithmically encoded images themselves suitable for conversion to intermediate imagery or output-referred imagery. The ability to defer demosaicing until some later time allows for recorded material to be processed with algorithms not available at the time or place of capture.

The image file structure encompasses a 4096-byte header and a one- or two-part payload. In an image file with a one-part payload, the entire payload is image data. In an image file with a two-part payload, one part carries the image data, and the other contains a one-dimensional lookup table [1D LUT] that can be used to implement a user-specified tonemapping operator during downstream processing.

The header is divided into subheaders each encompassing a particular type of image metadata. The subheaders and subheader fields that are defined in this document are sufficient to describe an array of sensor photosite data that has been subjected to an approximately logarithmic encoding function. This encoded photosite data can be converted to an intermediate image, or to an output-referred image suitable for final viewing. Subheaders and subheader fields marked as ‘reserved’ either hold metadata used for purposes other than logarithmic image construction, are reserved for compatible extensions of the documented file structure, or are unused.

When the image contains a two-part payload, the two parts can appear in either order (i.e., the 1D LUT indicating a user-specified tone mapping operator might precede the image data, or it might follow the image data). The header establishes the order implicitly by specifying the offset from the start of the file to the image data and (when present) the 1D LUT for the tonemapping operator.

Conversion of the encoded photosite data to any intermediate or to any output-referred image suitable for final viewing is outside the scope of this document. One such conversion is described in SMPTE RDD 31, Deferred Demosaicing of an ARRIRAW Image File to a Wide-Gamut Logarithmic Encoding.

It is the intent of this RDD to document the structure and interpretation of ARRIRAW files generated by ARRI cameras, so that users of this document may develop applications correctly identifying and interpreting such files. This document is specifically not intended to support development of hardware or of software applications creating ARRIRAW files. Permission to create such files, along with additional documentation to support that creation, is reserved by ARRI to members of the ARRI Partner Program, the contact information for which is provided below.

Use of this document to produce ARRIRAW files from non-ARRI cameras would potentially cause user confusion, diminished image quality as seen by content consumers, and damage of the reputation of the ARRIRAW brand and of ARRI itself.

Assistance in correctly processing ARRIRAW files (and all other types of files produced by ARRI cameras), including certification that the results of processing meet ARRI quality standards, is available to members of the ARRI Partner Program. Information on the ARRI Partner Program is available at the address given below:

Digital Workflow Solutions group – ARRI Partner Program
Arnold & Richter Cine Technik GmbH & Co. Betriebs (KG)
Turkenstraße 89
D-80799 München
Germany

Email: dws@arri.de

1 Scope

This document describes ARRIRAW files, defining an image file structure and an associated image data and image metadata interpretation that allows storage of photomosaic data from ARRI cameras, along with algorithms for processing that stored photomosaic data.

2 Normative References

The following standards contain provisions that, through reference in this text, constitute provisions of this registered disclosure document. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this registered disclosure document are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below.

IEEE Standard for Floating-Point Arithmetic P754-2008

ISO 12232:2006, Photography — Digital Still Cameras — Determination of Exposure Index, ISO Speed Ratings, Standard Output Sensitivity, and Recommended Exposure Index

Recommendation ITU-R BT.709 (04/2002), Parameter Values for the HDTV Standards for Production and International Programme Exchange

Recommendation ITU-R BT.1886 (03/2011), Reference Electro-Optical Transfer Function for Flat Panel Displays Used in HDTV Studio Production

SMPTE RDD 31:2014, Deferred Demosaicing of an ARRIRAW Image File to a Wide-Gamut Logarithmic Encoding

SMPTE ST 330:2011, Unique Material Identifier (UMID)

SMPTE RP 431-2:2011, D-Cinema Quality — Reference Projector and Environment

The Unicode Consortium. *The Unicode Standard, Version 6.3.0*, (Mountain View, CA: The Unicode Consortium, 2013. ISBN 978-1-936213-08-5)

3 Terms and Definitions

3.1
Base 36

Numerical encoding of quantities with radix 36, where the symbols representing the digits corresponding to decimal numbers 0 through 25 are the upper-case Latin characters A through Z and the symbols representing the digits corresponding to decimal numbers 26 through 35 are the decimal digits 0 through 9.

3.2

Null or Field Size Terminated String

Sequence of U8 data whose termination is indicated either by a NUL character or by the number of non-NUL characters in the sequence being the same as the size (in bytes) of the field defined to hold that sequence. When a NUL character terminates the sequence, all subsequent U8 elements in the sequence are also NUL characters.

4 Acronyms and Abbreviations

Acronyms used in this document are listed below.

ACES: Academy Color Encoding Specification

ASC: American Society of Cinematographers

CCT: Correlated Color Temperature

CDL: Color Decision List

DCI: Digital Cinema Initiatives

DT: Date and Time

EL: Encoder Limit

EOTF: Electro-Optical Transfer Function

GPS: Global Positioning System

IR: Infrared

LDS: Lens Data System

LAT: Latitude

LNG: Longitude

LTC: Linear Time Code

LUT: Lookup Table

N/A: Not Applicable

ND: Neutral Density

NFSTS: Null or Field Size Terminated String

NUL: UTF-8 character whose value is 0x00.

OCN: Original Camera Negative

P3: RGB primary chromaticities defined by DCI

R32: 32-bit IEEE 754-compliant floating-point number

REV: Raw Encoder Value

RGB: Red, Green and Blue

SA: Sensor Area

SUP: Software Update Packet

TC: Time code

U8: Unsigned 8-bit integer

U16: Unsigned 16-bit integer

U32: Unsigned 32-bit integer

U64: Unsigned 64-bit integer

UMID: Unique Media Identifier

UTF-8: Variable-width encoding that can encode every character in the Unicode character set

UUID: Universally Unique Identifier

UV: Ultraviolet

VFX: Visual Effects

VITC: Vertical Interval Time Code

5 Trademarks

ALEXA is a registered trademark of Arnold & Richter Cine Technik GmbH & Co.

Rosco is a registered trademark of Rosco Laboratories.

6 Definition of ARRIRAW Image File Structure, Image Metadata and Image Data

The image file structure comprises image metadata in the form of a header itself divided into 10 subheaders, followed by an image data area and (optionally) an area storing a user-supplied tonemap as a 1D LUT. The 10 subheaders are present in the image file, without any intervening data, in the order of their presentation in Sections 6.1.1 through 6.1.10. If the tonemapping 1D LUT is present, it either follows or precedes the image data area.

6.1 ARRIRAW Image Metadata Header Structure

The first byte of the aggregate header is the first byte of the image file. The length in bytes of the aggregate header is 4096 bytes.

Except for the Root subheader, each subheader is conceptually divided into two sections: a section aggregating individual fields of defined content and purpose, and a section comprised of a single field, the purpose of which is to reserve space within the subheader for future expansion. In the subheader descriptions below, this latter section, comprised of a single field, always has the label ‘Reserved’. The boundary between the two sections is indicated in each subheader by a field labeled Subheader Defined Content Length.

This scheme allows for the addition of new fields to the ARRIRAW image metadata header structure while maintaining the grouping of information for a like purpose. Expansion is accomplished by moving the boundary between the two sections (that is, by incrementing the Subheader Defined Content Length field for the subheader being augmented with new fields, and by equally decrementing the length of the Reserved field), with the newly defined fields appearing just before the relocated boundary.

The combined length of the individually defined content areas and the reserved area is appended to the title of the table describing the subheader structure.

Fields within a subheader are defined by their offset from the start of the subheader, and by their data type. Multi-byte data types are stored in little-endian order.

With two exceptions, if all bits in a subheader field are set, this indicates the field is unset and its meaning is undefined. Undefined content in a U32 field, for example, would have value 0xFFFFFFFF. The two exceptions are focal length and lens focus distance. In addition to being noted here, this is documented in the relevant section (Section 6.1.5).

Sequences of U8 data in some fixed-width fields, noted below, represent strings whose elements are interpreted as UTF-8 characters. The data type of such sequences is noted as being of type NFSTS, indicating a NUL or Field-Size Terminated String, further details of which are given in the Terms and Definitions section above.

6.1.1 Root Information Subheader Structure

Offset	Data Type	Count	Name	Content
0	U32	1	Magic Number	Verifies correct reading of multi-byte data types; value is 0x49525241. Ordered from least to most significant byte within this 32-bit unsigned integer, these bytes have UTF-8 character interpretations 'A', 'R', 'R' and 'I'.
4	U32	1	Byte Order	Verifies little-endianness of header; value is 0x78563412.
8	U32	1	Header Size	Length of aggregate header; value is 4096.
12	U32	1	Header Version	Version of aggregate header; value is 3.

6.1.2 Image Data Information Subheader Structure

The Image Data Information Subheader Structure is 68 bytes long, with 60 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	1	Width	Total horizontal pixel count of the stored photosite data
8	U32	1	Height	Total vertical pixel count of the stored photosite data.
12	U32	1	Subheader Version	Indicator of version of Image Data Information subheader; value is 1.
16	U32	1	Data Space	Indicator of encoding function used to store 16-bit photosite data as 12-bit raw Bayer pattern data in image area; value is 1 for data encoded as described in Section 6.2.2.
20	SA	1	Active Image Area	Rectangle defined on stored photosite data, representing the area that, after RGB reconstructions, is passed to downstream processing. Any image resizing requested by downstream consumers is done solely using the reconstructed RGB pixels indicated by this Active Image Area.
36	SA	1	Full Image Area	Rectangle defined on stored photosite data, representing the union of the area that is passed downstream (the 'Active Image Area') and any extra photosite data that might be useful in reconstructing RGB pixels within the Active Image Area. Photosites that are within the rectangle defined by the Width and Height fields but which are outside the Full Image Area are not supported by any in-camera dead pixel correction. The Full Image Area is always a proper superset of the Active Image Area.
52	U32	1	Image Data Offset	Offset from start of file to stored photosite data.
56	U32	1	Image Data Size	Size in bytes of stored photosite data.
60	U32	2	Reserved	Reserved for future use.

There are two ways in which a photosite's value might be used. Firstly, the photosite's value might be used to create a pixel that is accessible by downstream image consumers. Secondly, the photosite's value might be used to create a pixel that is accessible by downstream image consumers, but the pixel that is 'in register' with the photosite might not itself be so accessible. An example would be a photosite inside the Full Image Area, outside the Active Image Area, but close enough to the Active Image Area so that it contributed to the RGB reconstruction of a pixel within the Active Image Area.

6.1.2.1 Sensor Area (SA) Subsubheader Structure

The two structure elements 'Active Image Area' and 'Full Image Area' above are subsubheader structures representing sensor area; their data type is shown as 'SA'. The content of sensor area subsubheader structures is as indicated below.

Offset	Data Type	Count	Name	Content
0	U32	1	Left	0-based coordinate, in stored photosite data array, designating the first column index into that data array that can be used to access a photosite within the desired rectangular area of sensor data. The range of valid column indices is [Left, Left + Width).
4	U32	1	Top	0-based coordinate, in stored photosite data array, designating the first row index into that data array that can be used to access a photosite within the desired rectangular area of sensor data. The range of valid row indices is [Top, Top + Height).
8	U32	1	Width	Width, in photosites, of a rectangular area defined on the stored photosite data array.
12	U32	1	Height	Height, in photosites, of a rectangular area defined on the stored photosite data array.

6.1.3 Image Content Information Subheader Structure

The Image Content Information Subheader Structure is 268 bytes long, with 224 bytes currently being defined as detailed in this section.

Without loss of generality, the Image Content Information Subheader Structure is intended to support a processing algorithm containing the following steps, from image acquisition to image display:

- Linearization of the logarithmically-encoded photosite data.
- CCT- and tint-specific white balance factor application to photosite data, with different factors being applied to red, green and blue photosite data.
- Demosaicing of white-balanced photosite data to produce reconstructed sensor RGB values.
- Conversion of sensor RGB values to an alternative RGB representation with defined primaries and white point, those primaries encompassing significantly more of the CIE *u′v′* chromaticity diagram at typical luminances than would the primaries associated with an ITU-R BT.709 or DCI-P3 encoding (a 'wide-gamut' RGB encoding).
- Logarithmic encoding of the wide-gamut RGB data in an exposure-index-dependent manner.
- Optional offsetting of the logarithmically encoded wide-gamut RGB data.

- Application of a 1D LUT to implement a tonemapping operator, either using a default tonemapping 1D LUT provided by the processing model or an optional user-provided tonemapping 1D LUT.

- Conversion of the tonemapped RGB data for display using a different set of RGB primaries (and possibly white point) using a 3x3 matrix, with provision for optional adjustment of saturation as part of the conversion.

- Application of a 1D LUT to compensate for the presumed gamma of the viewing display.

- Optional application of ASC CDL slope, offset and power operations, in that order, to the display-ready data.

In the remainder of this document this will be referred to as the ‘nominal processing algorithm’.

The presence of fields in this subheader that record parameter values that could be used in the nominal processing algorithm described above does not preclude the effective use of the image data in other processing algorithms. A consumer of the image data could, for example, ignore all of the fields in this subheader and process the photosite data from start to finish in an ACES workflow if they so chose.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	1	Color Processing Version	Version number of color processing algorithm intended to be applied to image data, with value 4 for all images whose layout and semantics are as described by this document and with value less than 4 for prior versions not described in this document.
8	U32	1	White Balance CCT	Correlated color temperature selected on camera at time of image capture, in degrees Kelvin of a blackbody radiator or CIE daylight simulator assumed to be the illuminant, for the purposes of white balance factor and color matrix calculations.
12	R32	1	Green/Magenta Tint	Tint factor expressing deviation from blackbody radiator or CIE daylight simulator loci, as selected on camera at time of image capture. Values range from -12.0 to +12.0. The units of green/magenta tint do not correspond to any SI standard. A unit increment of green/magenta tint, however, is approximately equivalent to the color shift produced by a Rosco ± 1/8 green filter, or by a Kodak CC035 green or magenta filter.
16	R32	1	White Balance Red Factor	Factor by which red photosite data are multiplied prior to application of demosaicing algorithm.
20	R32	1	White Balance Green Factor	Factor by which green photosite data are multiplied prior to application of demosaicing algorithm. The value of this field is always 1.0.
24	R32	1	White Balance Blue Factor	Factor by which blue photosite data are multiplied prior to application of demosaicing algorithm.

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>28</td><td>U32</td><td>1</td><td>White Balance Baking</td><td>Indicator that stored image photosite data values have been multiplied by the white balance factors prior to storage in the image data area.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>indicates that no such multiplication has yet been done to the image data.</td></tr><tr><td>1</td><td>indicates that the image data have been so multiplied.</td></tr></table>The value of this field is 0 for all images whose layout and semantics are as described by this document.</td></tr>
<tr><td>32</td><td>U32</td><td>1</td><td>Exposure Index</td><td>Effective exposure index selected on camera at time of image data capture.</td></tr>
<tr><td>36</td><td>U32</td><td>1</td><td>Black Level</td><td>In prior versions of this header, code value corresponding to black level. Obsolete and unset.</td></tr>
<tr><td>40</td><td>U32</td><td>1</td><td>White Level</td><td>In prior versions of this header, code value corresponding to white level. Obsolete and unset.</td></tr>
<tr><td>44</td><td>R32</td><td>12</td><td>Color Matrix</td><td>3 x 4 matrix by which, in the nominal processing algorithm, RGB values resulting from demosaicing are multiplied. The matrix is applied to the left of a 4-element column vector whose values are the red, green and blue results of the demosaicing algorithm along with a unity value placed in the bottom row of the column vector. The contents of the 3 rows are stored in top-to-bottom order, with the contents of each row stored in left-to-right order; the last column of the matrix represents an offset.</td></tr>
<tr><td>92</td><td>R32</td><td>1</td><td>Color Matrix Desaturation Gain</td><td>Gain value of the color matrix desaturation range. Obsolete.</td></tr>
<tr><td>96</td><td>R32</td><td>1</td><td>Color Matrix Desaturation Offset</td><td>Offset value of the color matrix desaturation range. Obsolete.</td></tr>
<tr><td>100</td><td>U32</td><td>1</td><td>Highlight Desaturation</td><td>Indicator of an obsolete form of highlight desaturation processing not described in this document.</td></tr>
<tr><td>104</td><td>U32</td><td>1</td><td>Target Color Space</td><td>Indicator of the color space into which, by default, this image data will be transformed when the nominal processing algorithm is used.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>indicates an output-referred color space with the same primary chromaticity coordinates and white point as ITU-R BT.709 and a presumed display device EOTF matching that described in ITU-R BT.1886.</td></tr><tr><td>1</td><td>indicates an output-referred color space with the same primary chromaticity coordinates and white point as that specified by SMPTE RP 431-2.</td></tr><tr><td>2</td><td>indicates the wide-gamut color space produced after the CCT-dependent matrix multiplication and subsequent logarithmic encoding.</td></tr><tr><td>3</td><td>indicates the reconstructed camera RGB values prior to the application of the CCT-dependent matrix but subsequently processed by the usual post-matrix logarithmic encoder.</td></tr><tr><td>4</td><td>indicates a color space approximating the printing densities that would be produced by optimally capturing the scene on OCN stock, optimally processing that material and optimally scanning the resulting OCN on an ARRISCAN scanner.</td></tr></table></td></tr>
</table>

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>108</td><td>U32</td><td>1</td><td>Sharpness</td><td>Indicator of the amount of sharpening to be applied to the image data in the nominal processing algorithm. The value is between 0 and 300 inclusive. There is neither a metric for sharpening strength nor any guarantee of perceptual linearity in perceived sharpness across the value range. The strength of sharpening between 0 and 300 does, however, increase monotonically.
<table>
<tr><td>Value</td><td>Interpretation</td></tr>
<tr><td>0</td><td>indicates no sharpening.</td></tr>
<tr><td>100</td><td>the default level of sharpening.</td></tr>
<tr><td>300</td><td>maximal sharpening.</td></tr>
</table></td></tr>
<tr><td>112</td><td>R32</td><td>1</td><td>Pixel Aspect Ratio</td><td>Factor by which reconstructed pixel data need to be stretched horizontally to compensate for anamorphic distortion and restore, within the limitations of the design of the anamorphic lens, a rectilinear relationship between objects in the scene and the image of the scene in reconstructed RGB pixels.</td></tr>
<tr><td>116</td><td>U32</td><td>1</td><td>Flip or Flop</td><td>Indicator of any top-to-bottom reversals (flips) or left-to-right reversals (flops) that are performed on image data for evaluative viewing.
<table>
<tr><td>Value</td><td>Interpretation</td></tr>
<tr><td>0</td><td>indicates no flipping or flopping is desired.</td></tr>
<tr><td>1</td><td>indicates any displayed image are flopped (horizontally reversed relative to original scene).</td></tr>
<tr><td>2</td><td>indicates any displayed image are flipped (vertically reversed relative to original scene).</td></tr>
<tr><td>3</td><td>indicates any displayed image are rotated 180º (that is, vertically and horizontally reversed relative to original scene).</td></tr>
</table></td></tr>
<tr><td>120</td><td>NFSTS</td><td>32</td><td>Look File Name</td><td>Name of the Look File. When the nominal processing algorithm is used, if the Look File Name is set to ‘None’, then processing is performed as if any Look File field was set to its default value, regardless of the actual value stored in that position in the image header.</td></tr>
<tr><td>152</td><td>U32</td><td>1</td><td>Look File Curve Type</td><td>Indicator as to presence or absence of user-supplied Look File Curve.
<table>
<tr><td>Value</td><td>Interpretation</td></tr>
<tr><td>0</td><td>indicates no Look File Curve present.</td></tr>
<tr><td>1</td><td>indicates Look File Curve is present.</td></tr>
</table></td></tr>
<tr><td>156</td><td>U32</td><td>1</td><td>Look File Curve Offset</td><td>Offset from start of file to Look File Curve data.</td></tr>
<tr><td>160</td><td>U32</td><td>1</td><td>Look File Curve Size</td><td>Size, in bytes, of Look File Curve data.</td></tr>
<tr><td>164</td><td>U32</td><td>1</td><td>Look File Curve CRC</td><td>In prior versions of this header, code value corresponding to checksum of look file curve. Obsolete and unset.</td></tr>
</table>

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>168</td><td>R32</td><td>1</td><td>Saturation</td><td>Factor by which the saturation of the image is altered in its conversion to the target color space, when the nominal color processing algorithm is used. The factor indicates a linear interpolation of matrix coefficients, between a black-and-white conversion matrix on the one hand and the normal target color space conversion matrix on the other.
<table>
<tr><td>Value</td><td>Interpretation</td></tr>
<tr><td>< 0.0</td><td>not allowed.</td></tr>
<tr><td>0.0</td><td>indicates full desaturation (black-and-white).</td></tr>
<tr><td>1.0</td><td>indicates normal saturation.</td></tr>
<tr><td>2.0</td><td>indicates maximum extrapolated saturation.</td></tr>
<tr><td>> 2.0</td><td>not allowed.</td></tr>
</table>
Values between 0.0 and 1.0 linearly interpolate matrix coefficients between that of a completely desaturating matrix and a matrix producing normal saturation. Values between 1.0 and 2.0 linearly extrapolate matrix coefficients between a matrix producing normal saturation and a matrix producing twice normal saturation.

If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 1.0.</td></tr>
<tr><td>172</td><td>RGB</td><td>1</td><td>CDL Slope</td><td>Triplet of values by which the red, green and blue channels of normalized RGB images is multiplied, when the nominal color processing algorithm is used.

If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 1.0 in each of its red, green and blue components.</td></tr>
<tr><td>184</td><td>RGB</td><td>1</td><td>CDL Offset</td><td>Triplet of values by which the red, green and blue channels of normalized RGB images is offset, when the nominal color processing algorithm is used.

If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 0.0 in each of its red, green and blue components.</td></tr>
<tr><td>196</td><td>RGB</td><td>1</td><td>CDL Power</td><td>Triplet of values to which power the red, green and blue channels of normalized RGB images is raised, when the nominal color processing algorithm is used.

If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 1.0 in each of its red, green and blue components.</td></tr>
<tr><td>208</td><td>RGB</td><td>1</td><td>Printer Light Adjustment</td><td>Normalized value by which image stored in the ALEXA V3 Log C encoding is offset prior to processing by the ALEXA tonemapping operator, when the nominal color processing algorithm is used.

If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 0.0 in each of its red, green and blue components.</td></tr>
</table>

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>220</td><td>U32</td><td>1</td><td>CDL Application Mode</td><td>Indicator of how Saturation adjustment and CDL Slope, Offset and Power adjustments are applied to image data, when the nominal color processing algorithm is used.
<table>
<tr><th>Value</th><th>Interpretation</th></tr>
<tr><td>0</td><td>indicates that no printer light adjustments are made prior to the application of the default or any user-supplied tonemapping LUT, no saturation adjustment are folded into the default conversion to display RGB, and no CDL Slope, Offset and Power operations are applied during color processing.</td></tr>
<tr><td>1</td><td>indicates that the default saturation processing is modified as described in the description of the Saturation field above.</td></tr>
<tr><td>2</td><td>indicates that instead of the value of the Saturation field being interpreted as an interpolation or extrapolation between a conversion to black-and-white on the one hand, and standard conversion to a display color space on the other, the Saturation field effects a saturation transformation as described in the ASC CDL document.</td></tr>
<tr><td>3</td><td>indicates that slope, offset, power and saturation are applied, in that order, to the logarithmic image data that is the result of the procedures described in this document.</td></tr>
</table>
If the value of the Look File Name field is ‘None’, then regardless of the actual content of this field in the header, processing proceeds as if this field were set to 0.</td></tr>
<tr><td>224</td><td>U8</td><td>44</td><td>Reserved</td><td>Reserved for future use.</td></tr>
</table>

6.1.3.1 RGB Operator Parameter (RGB) Subsubheader Structure

Offset	Data Type	Count	Name	Content
0	R32	1	Red	Parameter indicating argument for operator on red component of pixel value.
4	R32	1	Green	Parameter indicating argument for operator on green component of pixel value.
8	R32	1	Blue	Parameter indicating argument for operator on blue component of pixel value.

6.1.4 Camera Device Information Subheader Structure

The Camera Device Information Subheader Structure is 528 bytes long, with 408 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	1	Camera Type ID	Number indicating camera type. 2 (indicating an ARRI ALEXA camera) for all images whose layout and semantics are as described by this document.

Offset	Data Type	Count	Name	Content
8	U32	1	Camera Revision	Number indicating camera hardware update level.
12	U32	1	Firmware Version	Number indicating installed camera firmware update level.
16	U32	1	Camera Serial Number	Camera serial number.
20	U32	1	Camera ID	Camera serial number in Base 36 (A-Z + 0-9).
24	U32	1	Camera Index	User-specified camera index, as a UTF-8 character, in the least-significant byte of this field.
28	DT	1	System Image Creation Time	Date and time image was created, in the time zone chosen during camera setup.
36	DT	1	User Defined Time	In prior versions of this header, user-defined date and time. Obsolete and unset.
44	U32	1	Exposure Time	Exposure time in microseconds.
48	U32	1	Shutter Angle	Shutter angle in thousandths of a degree.
52	U32	1	Speed Ramp Duration	In prior versions of this header, duration of speed ramp in frames. Obsolete and unset.
56	U32	1	Speed Ramp Start Frame	In prior versions of this header, flag to indicate this frame is the speed ramp start frame. Obsolete and unset.
60	U32	1	Speed Ramp End Frame	In prior versions of this header, flag to indicate this frame is the speed ramp end frame. Obsolete and unset.
64	U32	1	Sensor Frames Per Second	Speed at which images are acquired, in thousandths of a frame per second.
68	U32	1	Project Frames Per Second	Speed at which images are intended to be played back, in thousandths of a frame per second.
72	TC	1	Master Timecode	TImecode.
88	TC	1	External LTC Free Run	External linear timecode.
104	TC	1	External VITC	External vertical interval timecode.
120	TC	1	Internal Free Run Time-of-day Timecode	Internal free run timecode, corresponding to time of day.
136	TC		Internal Free Run User-Defined Timecode	Internal free run timecode, with user-defined semantics.
152	TC		Recorder Run Edge Code Timecode	Auxiliary timecode, increasing with each insertion of new recording media, generated from media frame counter.
168	TC		Recorder Run Clip Code TImecode	Auxiliary timecode, increasing with each new take, generated from take frame counter.
184	TC		Recorder Run Regen Timecode	Internal recorder run regen timecode, with user-defined semantics.
200	TC	4	Reserved Timecode	Reserved for future types of timecode.
264	U64	1	Magazine Serial Number	Serial number of magazine used to capture this frame.
272	U32	3	Digital Keycode	In prior versions of this header, digital keycode. Obsolete and unset.
284	U8	32	SMPTE UMID	SMPTE ST 330 UMID identifying instances of audiovisual material.
316	NFSTS	8	Camera Type	Name of type of camera.
324	NFSTS	32	Recorder Type	Name of type of recorder, or type of storage media.

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>356</td><td>U32</td><td>1</td><td>Device Information Flags</td><td>Collection of flags indicating miscellaneous camera state information.
<table>
<tr><td>Position</td><td>Name</td><td>Value</td><td>Meaning</td></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Mirror Shutter</td><td>0</td><td>Camera had mirror shutter and mirror shutter was active when image data were captured.</td></tr>
<tr><td>1</td><td>Camera either did not have mirror shutter, or had mirror shutter but mirror shutter was inactive when image data were captured.</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Variframe</td><td>0</td><td>Variframe was inactive, or variframe was active and this is a valid image.</td></tr>
<tr><td>1</td><td>Variframe was active and this is a duplicate image.</td></tr>
<tr><td>2..31</td><td>Reserved</td><td></td><td>Reserved for future use.</td></tr>
</table></td></tr>
<tr><td>360</td><td>R32</td><td>1</td><td>Exposure phase</td><td>Phase between center of exposure and center of timecode, in percent of frame duration.</td></tr>
<tr><td>364</td><td>U32</td><td>1</td><td>Sub-second Frame Count</td><td>Zero-based count of frame within the current second’s worth of timecode, in support of frame rates higher than those which can be encoded in SMPTE timecode.</td></tr>
<tr><td>368</td><td>U8</td><td>16</td><td>ARRI_UUID</td><td>UUID created by camera for each clip.</td></tr>
<tr><td>384</td><td>NFSTS</td><td>24</td><td>SUP name</td><td>Name indicating installed camera firmware update level.</td></tr>
<tr><td>408</td><td>U32</td><td>30</td><td>Reserved</td><td>Reserved for future use.</td></tr>
</table>

Camera serial numbers are assigned in large blocks to different models of the ARRI ALEXA camera, and therefore a higher camera serial number does not in itself signify a more recent date of manufacture.

Neither a time zone nor any indication of a daylight savings time offset is stored in the file. The time is stored relative to whatever time zone the camera operator designates during camera setup. This might be the time zone local to the place of capture, or some other time zone, according to the wishes of the operator.

6.1.4.1 Date and Time (DT) Subsubheader Structure

Offset	Data Type	Count	Name	Content
0	U32	1	Date	Date as YYYYMMDD, with each digit represented in 4 bits, with YYYY stored in the most significant two bytes, DD in the least-significant byte and MM in the intervening byte.
4	U32	1	Time	Time as HHMMSScc, with each digit represented in 4 bits, with HH stored in the most significant two bytes, cc (centiseconds) in the least-significant byte and MM and SS in the intervening two bytes.

6.1.4.2 Timecode (TC) Subsubheader Structure

Offset	Data Type	Count	Name	Content
0	U32	1	Timecode	Timecode as binary-coded decimal value, with each digit represented in four bits, in HH:MM:SS:FF order, with HH stored in the most significant byte, FF in the least-significant byte, and MM and SS in the intervening two bytes.
4	U32	1	Frame Counter	Frame counter value derived from timecode and timebase.
8	U32	1	Timebase	Timebase in thousandths of a frame per second.
12	U32	1	Drop Frame	Indicator for drop-frame timecode. Value – Interpretation 0 – indicates the timecode is non-drop-frame timecode to be interpreted as HH:MM:SS:FF. 1 – indicates the timecode is drop-frame timecode to be interpreted as HH:MM:SS;FF.

6.1.5 Lens Data Information Subheader Structure

The Lens Data Information Subheader Structure is 200 bytes long, with 116 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Use
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	1	Focus unit	Unit of focus for focus distance. Value – Interpretation 0 – Focus distance is in thousandths of an inch. 1 – Focus distance is in microns.
8	U32	1	Focus Distance	Distance in the specified focus unit, in thousandths of an inch or in microns. If 0, focus distance is invalid or unknown. Note: The use of 0xFFFFFFFF in a field to indicate that the field's contents are unset does not apply to this field. In the case that this field is set to 0xFFFFFFFF, the field is taken to indicate that the focus distance is infinite.
12	U32	1	Focal Length	Focal length in microns, or 0 if no focal length is known. Note: The use of 0xFFFFFFFF in a field to indicate that the field's contents are unset does not apply to this field. In the case that this field is set to 0xFFFFFFFF, the field is taken to indicate that the focal length is unknown.
16	U32	1	Lens Serial Number	Lens Serial Number.
20	U32	1	Aperture	Aperture in thousandths in linear range, i.e. log of optical aperture indicated on lens ring.
24	ND	1	ND Filter	Subsubheader structure indicating use of ND filter.
28	U32	1	UV Filter	Indicator for ultraviolet filter use. Value – Interpretation 0 – No UV filter was used. 1 – A UV filter was used.

<table>
<tr><td>32</td><td>U32</td><td>1</td><td>IR Filter</td><td>Indicator for infrared filter use.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>No IR filter was used.</td></tr><tr><td>1</td><td>An IR filter was used.</td></tr></table></td></tr>
<tr><td>36</td><td>U32</td><td>1</td><td>IR Black Filter</td><td>Indicator for infrared black filter use.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>No IR black filter was used.</td></tr><tr><td>1</td><td>An IR black filter was used.</td></tr></table></td></tr>
<tr><td>40</td><td>NFSTS</td><td>32</td><td>Lens Model</td><td>Name of lens model.</td></tr>
<tr><td>72</td><td>REV</td><td>1</td><td>Focus Distance Raw Encoder Value</td><td>Uncalibrated encoder value for focus distance.</td></tr>
<tr><td>76</td><td>REV</td><td>1</td><td>Focal Length Raw Encoder Value</td><td>Uncalibrated encoder value for focal length.</td></tr>
<tr><td>80</td><td>REV</td><td>1</td><td>Aperture Raw Encoder Value</td><td>Uncalibrated encoder value for aperture.</td></tr>
<tr><td>84</td><td>EL</td><td>1</td><td>Focus Distance LDS Encoder Limits</td><td>Range of calibrated, LDS-derived focus distance values.</td></tr>
<tr><td>88</td><td>EL</td><td>1</td><td>Focal Length LDS Encoder Limits</td><td>Range of calibrated, LDS-derived focal length values.</td></tr>
<tr><td>92</td><td>EL</td><td>1</td><td>Aperture LDS Encoder Limits</td><td>Range of calibrated, LDS-derived aperture values.</td></tr>
<tr><td>96</td><td>EL</td><td>1</td><td>Focus Distance Motor Encoder Limits</td><td>Range of calibrated, motor-derived focus distance values.</td></tr>
<tr><td>100</td><td>EL</td><td>1</td><td>Focal Length Motor Encoder Limits</td><td>Range of calibrated, motor-derived focal length values.</td></tr>
<tr><td>104</td><td>EL</td><td>1</td><td>Aperture Motor Encoder Limits</td><td>Range of calibrated, motor-derived aperture values.</td></tr>
<tr><td>108</td><td>U16</td><td>1</td><td>LDS Lag Type</td><td>Indicator of interpretation of calibrated LDS lag value<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>Undefined (0xFFFF)</td><td>LDS lag value is invalid.</td></tr><tr><td>0</td><td>LDS lag value is invalid.</td></tr><tr><td>1</td><td>LDS lag value is valid.</td></tr></table></td></tr>
<tr><td>110</td><td>U16</td><td>1</td><td>LDS Lag Value</td><td>Lag of calibrated LDS data, in frames, behind the image data gathered at the same time the calibrated LDS value was gathered.</td></tr>
<tr><td>112</td><td>U16</td><td>1</td><td>Raw Encoder Lag Type</td><td>Indicator of interpretation of raw encoder lag value<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>Undefined (0xFFFF)</td><td>raw encoder lag value is invalid.</td></tr><tr><td>0</td><td>raw encoder lag value is invalid.</td></tr><tr><td>1</td><td>raw encoder lag value is valid.</td></tr></table></td></tr>
<tr><td>114</td><td>U16</td><td>1</td><td>Raw Encoder Lag Value</td><td>Lag of stored uncalibrated LDS data, in frames, behind the image data gathered at the same time the stored uncalibrated LDS value was gathered.</td></tr>
<tr><td>116</td><td>U32</td><td>21</td><td>Reserved</td><td>Reserved for future use.</td></tr>
</table>

6.1.5.1 Neutral Density (ND) Subsubheader Structure

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Use</th></tr>
<tr><td>0</td><td>U16</td><td>1</td><td>Filter Type</td><td>Indicator of ND filter type used.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>No ND black filter was used.</td></tr><tr><td>1</td><td>An ALEXA Studio ND Type 1 filter was used.</td></tr></table></td></tr>
<tr><td>2</td><td>U16</td><td>1</td><td>Filter Density</td><td>Optical density of ND filter in thousandths.</td></tr>
</table>

6.1.5.2 Raw Encoder Value (REV) Subsubheader Structure

Offset	Data Type	Count	Name	Use
0	U16	1	LDS Encoder Value	Uncalibrated LDS encoder value.
2	U16	1	Motor Encoder Value	Uncalibrated motor encoder value.

6.1.5.3 Encoder Limits (EL) Subsubheader Structure

Offset	Data Type	Count	Name	Use
0	U16	1	Minimum Encoder Limit	Lower limit on calibrated LDS or motor encoder value.
2	U16	1	Maximum Encoder Limit	Upper limit on calibrated LDS or motor encoder value.

6.1.6 VFX Information Subheader Structure

The VFX Information Subheader Structure is 184 bytes long, with 56 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Use
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	LAT	1	GPS Latitude	Camera latitude at time of image capture.
12	LNG	1	GPS Longitude	Camera longitude at time of image capture.
20	U32	1	Camera X	Camera position, in units of a thousandth of a meter, along the X axis of some Cartesian coordinate system. Not currently implemented.
24	U32	1	Camera Y	Camera position, in units of a thousandth of a meter, along the Y axis of some Cartesian coordinate system. Not currently implemented.
28	U32	1	Camera Z	Camera position, in units of a thousandth of a meter, along the Z axis of some Cartesian coordinate system. Not currently implemented.
32	U32	1	Camera Pan	Camera pan, in units of a tenth of a degree, relative to user-established external reference. Not currently implemented.
36	U32	1	Camera Tilt	Camera tilt, in units of a tenth of a degree, from last level calibration, where a value of +900 means ‘straight up’ and a value of -900 means ‘straight down’. Though the field is defined as an unsigned 32-bit integer the camera produces 16-bit signed values stored in the least-significant 16 bits of the field.
40	U32	1	Camera Roll	Camera roll, in units of a tenth of a degree, from last level calibration, where a value of 1800 means clockwise rotation sufficient to produce an inverted image, and a value of -1800 means counter-clockwise rotation to similarly produce an inverted image. Though the field is defined as an unsigned 32-bit integer the camera produces 16-bit signed values stored in the least-significant 16 bits of the field. Positive values represent clockwise rotation as seen from the perspective of someone standing behind the camera.

<table>
<tr><td>44</td><td>U32</td><td>1</td><td>Master Flag</td><td>Indicator of role in multi-camera shoot.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>camera was shooting independently.</td></tr><tr><td>1</td><td>camera was acting as master in a multi-camera shoot.</td></tr><tr><td>2</td><td>camera was acting as slave in a multi-camera shoot.</td></tr></table></td></tr>
<tr><td>48</td><td>U32</td><td>1</td><td>Image Channel</td><td>Indicator of role in multi-camera shoot.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>image is from a single-camera shoot.</td></tr><tr><td>1</td><td>image is from left eye of a stereo shoot.</td></tr><tr><td>2</td><td>image is from right eye of a stereo shoot.</td></tr></table></td></tr>
<tr><td>52</td><td>U32</td><td>1</td><td>Partner ID</td><td>Camera ID from other camera in 3D configuration, if Channel Info value is 1 or 2.</td></tr>
<tr><td>56</td><td>U32</td><td>32</td><td>Reserved</td><td>Reserved for future use.</td></tr>
</table>

This RDD does not mandate a particular Cartesian coordinate system in which Camera X, Camera Y and Camera Z distances from an origin would be defined.

This RDD does not mandate any particular nesting order of pan, tilt or roll angles.

6.1.6.1 GPS Latitude (LAT) Subsubheader Structure

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>0</td><td>U32</td><td>1</td><td>North</td><td>Indicator if this is a latitude in the northern hemisphere.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>the latitude is interpreted as being in the northern hemisphere, with 90 meaning at the north pole and 0 meaning at the equator.</td></tr><tr><td>1</td><td>the latitude is interpreted as being in the southern hemisphere, with 0 meaning at the equator and -90 meaning at the south pole.</td></tr></table></td></tr>
<tr><td>4</td><td>U32</td><td>1</td><td>Latitude</td><td>Latitude in ten-millionths of a decimal degree.</td></tr>
</table>

6.1.6.2 GPS Longitude (LNG) Subsubheader Structure

<table>
<tr><th>Offset</th><th>Data Type</th><th>Count</th><th>Name</th><th>Content</th></tr>
<tr><td>0</td><td>U32</td><td>1</td><td>East</td><td>Indicator if this is an eastern longitude.<table><tr><td>Value</td><td>Interpretation</td></tr><tr><td>0</td><td>the longitude is interpreted as being east of the prime meridian, with longitudinal values varying between 0 and 180 degrees.</td></tr><tr><td>1</td><td>the longitude is interpreted as being west of the prime meridian, with longitudinal values varying between 0 and -180 degrees.</td></tr></table></td></tr>
<tr><td>4</td><td>U32</td><td>1</td><td>Longitude</td><td>Longitude in ten-millionths of a decimal degree.</td></tr>
</table>

6.1.7 Clip Information Subheader Structure

The Clip Information Subheader Structure is 552 bytes long, with 448 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	1	Circle Take	Indicator that take was ‘circled’. Value: Interpretation 0: the take was not considered a circled take. 1: the take was considered a circled take.
8	NFSTS	8	Reel Name	Name of the virtual ‘reel’ on which this frame was recorded.
16	NFSTS	16	Scene Name	Name of the scene being captured when this frame was recorded.
32	NFSTS	8	Take Name	Name of the take being captured when this frame was recorded.
40	NFSTS	32	Director	Name of the director of the production of which this frame is a part.
72	NFSTS	32	Director of Photography	Name of the cinematographer directing the recording of which this frame is a part.
104	NFSTS	32	Production Name	Name of the production of which this frame is a part.
136	NFSTS	32	Company Name	Name of the company producing the content of which this frame is a part.
168	NFSTS	256	Notes	Free-form text containing any notes from the capture of the clip of which this frame is a part.
424	NFSTS	24	Camera Clip Name	Name of the clip of which this frame is a part, as generated by the camera.
448	U32	26	Reserved	Reserved for future use.

6.1.8 Sound Information Subheader Structure

The Sound Information Subheader Structure is 220 bytes long, with 180 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	TC	1	Sound Timecode	Analog sound timecode from external sound device.
20	NFSTS	32	Sound File Name	Sound file name.
52	NFSTS	32	Sound Roll Name	Sound roll name.
84	NFSTS	32	Scene File Name	External audio scene file name.
116	NFSTS	32	Take File Name	External audio take file name.
148	NFSTS	32	User-Defined Info	User-defined information string.
180	U32	10	Reserved	Reserved for future use.

6.1.9 Camera Information Subheader Structure

The Camera Information Subheader Structure is 84 bytes long, with 40 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U32	9	Component Information	Proprietary and camera-model-specific data describing internal hardware state at time of capture.
40	U32	11	Reserved	Reserved for future use.

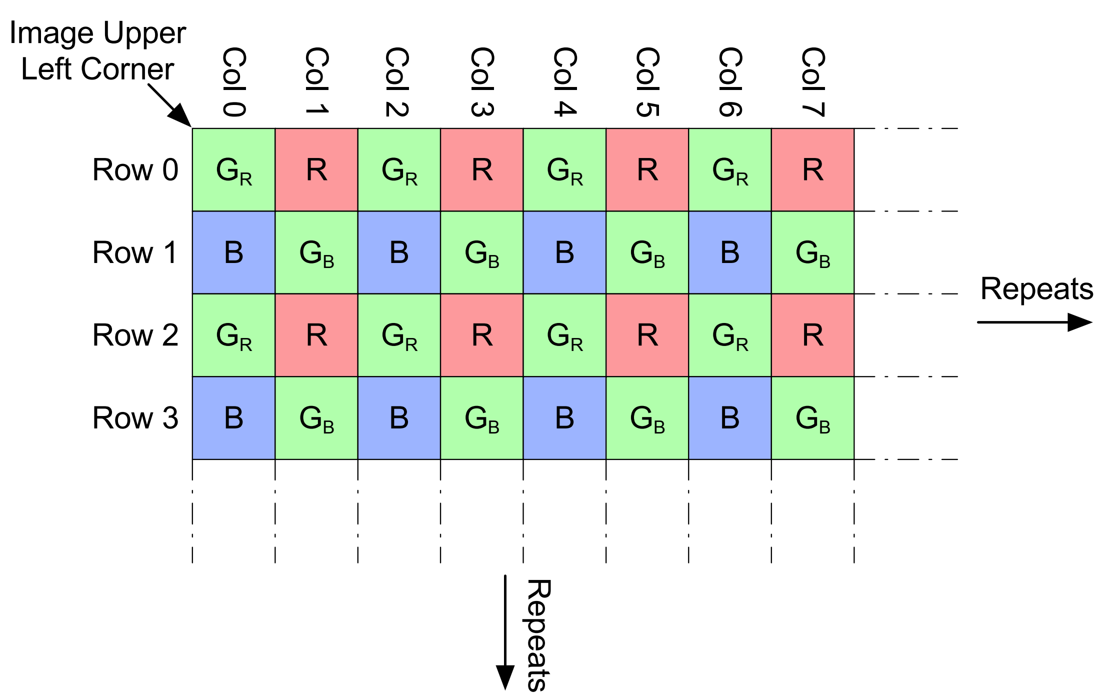
6.1.10 ARRI-Reserved Subheader Structure

The ARRI-Reserved Subheader Structure is 1976 bytes long, with 4 bytes currently being defined as detailed in this section.

Offset	Data Type	Count	Name	Content
0	U32	1	Subheader Defined Content Length	Length of the defined content of this subheader in bytes, including this length field.
4	U8	1972	Reserved	Reserved for future use.

6.2 ARRIRAW Image Data

The image data represent a 12-bit logarithmic encoding of 16-bit photosite values at time of capture, packed into 32-bit words in the data portion of the image file. There are four types of photosites: $\text{Green}_{\text{Red}}$, Red, Blue, and $\text{Green}_{\text{Blue}}$. Rows of photosites are either alternations of $\text{Green}_{\text{Red}}$ and Red, or of Blue and $\text{Green}_{\text{Blue}}$. A row containing $\text{Green}_{\text{Red}}$ and Red photosites will begin with a $\text{Green}_{\text{Red}}$ photosite. A row containing Blue and $\text{Green}_{\text{Blue}}$ photosites will begin with a Blue photosite. The leftmost, uppermost photosite is a $\text{Green}_{\text{Red}}$ photosite. This leads to the following layout of photosites:



6.2.1 Sensor traversal order

The data are stored in top-down row-scan order: the first datum represents the value of the leftmost uppermost photosite (a $\text{Green}_{\text{Red}}$ photosite) and successive values traverse the sensor in row-major order; i.e., the column varies fastest (from leftmost to rightmost column) and the row varies slowest (from topmost to bottommost row).

6.2.2 12-bit Approximately Logarithmic Storage Encoding of Sensor Data

Photosite values are radiometrically linear representations of the energy they receive, represented as 16-bit unsigned integers, incorporating an offset such that a photosite receiving no energy would have a 16-bit unsigned integer value of 256.

Before being stored as file image data, they are encoded using a method that approximates a logarithmic encoding, but emphasizes the preservation of deep shadow detail.

The following equation encodes a 16-bit linear integer photosite value v_p as an encoded 12-bit integer value v_i.

$$v_i = \begin{cases} v_p, & v_p < 1024 \\ 512 \times q + (v_p \gg q) & v_p \geq 1024 \end{cases}$$

where

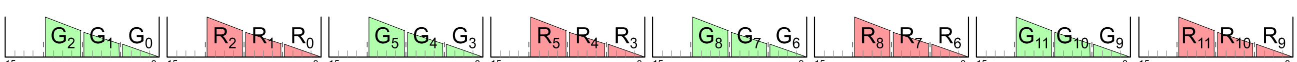
q is the integer part of the difference $\log_2(v_p) - 9$

6.2.3 Packing of 12-bit Encoded Values into 32-bit Words

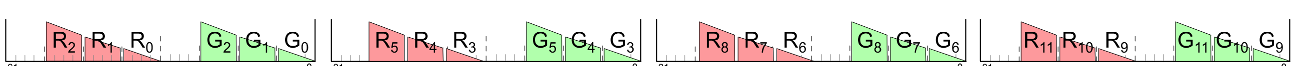
The equation presented in Section 6.2.2 takes a 16-bit linear integer value and produces a 12-bit near-logarithmic integer value. Subsequent to this encoding a sequence of 12-bit near-logarithmic integer values is packed into a sequence of 32-bit words, prior to being written to a file as a sequence of 8-bit bytes as will be described in Section 6.2.4.

The algorithm presented below assumes that the encoding of 16-bit integer photosite values occurs in sensor traversal order to produce a sequence of 12-bit integer values stored in the least significant 12 bits of 16-bit half-words; that successive 16-bit half-words are stored adjacently in memory, and that the packing algorithm is executed on a machine with a little-endian architecture.

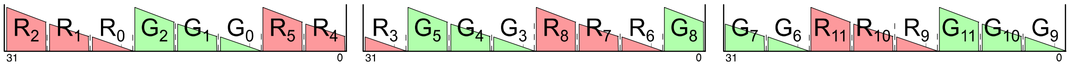
The illustration below shows an example of how such 12-bit integer values would be organized in memory as consecutive 16-bit half-words, with the example data taken (without loss of generality) from a row of $\text{Green}_{\text{Red}}$ and Red photosites. Within each 12-bit integer value, the most significant 4 bits are shown as a larger trapezoid, the next-most-significant 4 bits are given as a smaller trapezoid, and the least significant 4 bits are shown as a triangle.



Interpretation of that same memory as 32-bit words, as is done by the algorithm given in Section 6.2.3.1, implies that the algorithm will appear to have a 16-bit value in its most significant 16 bits corresponding to a photosite that precedes, in sensor traversal order, the photosite whose 16-bit value is in its least significant 16 bits, as shown in the following illustration.



The packing algorithm whose pseudocode is given in Section 6.2.3.1.2 takes as input blocks of 4 32-bit words and producing as output blocks of 3 32-bit words. Such an output block is illustrated below.



6.2.3.1 Pseudocode –Packing 12-bit Encoded Values Into 32-bit Words

6.2.3.1.1 Variable definitions

`NumPixel`	number of pixel in the Bayer pattern image
`InSize`	memory used to store the 12-bit image
`OutSize`	memory size required to store the packed image
`lpIn`	pointer to the input packed image memory
`lpOut`	pointer to the output packed image memory
`MSB_Pixel`	pixel in the 16 most significant bits of the output 32-bit memory word
`LSB_Pixel`	pixel in the 16 least significant bits of the output 32-bit memory word
`OutWord`	output 32-bit packed word

6.2.3.1.2 Pseudocode implementation

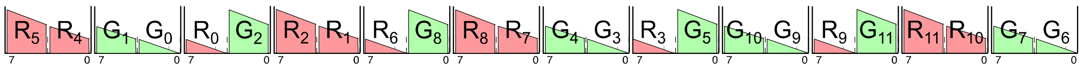
```
void ARI_Pack12BitBayerPattern(uint32 *lpIn, uint32 *lpOut, ...)

{

        OutSize = 12*InSize/16;

        While (NumPixel > 0)

        {

                MSB_Pixel = ((lpIn[0] >> 16) & 0x0000FFFF);

                LSB_Pixel = (lpIn[0] & 0x0000FFFF);

                OutWord = (MSB_Pixel << 20) & 0xFFF00000;

                OutWord |= ((LSB_Pixel << 8) & 0x000FFF00);

                MSB_Pixel = ((lpIn[1] >> 16) & 0x0000FFFF);

                LSB_Pixel = (lpIn[1] & 0x0000FFFF);

                OutWord |= ((MSB_Pixel >> 4) & 0x000000FF);
```

```
            lpOut[0] = OutWord;
            OutWord = ((MSB_Pixel << 28) & 0xF0000000);
            OutWord |= ((LSB_Pixel << 16) & 0x0FFF0000);
            MSB_Pixel = ((lpIn[2] >> 16) & 0x0000FFFF);
            LSB_Pixel = (lpIn[2] & 0x0000FFFF);
            OutWord |= ((MSB_Pixel << 4) & 0x0000FFF0);
            OutWord |= ((LSB_Pixel >> 8) & 0x0000000F);
            lpOut[1] = OutWord;
            OutWord = ((LSB_Pixel << 24) & 0xFF000000);
            MSB_Pixel = ((lpIn[3] >> 16) & 0x0000FFFF);
            LSB_Pixel = (lpIn[3] & 0x0000FFFF);
            OutWord |= ((MSB_Pixel << 12) & 0x00FFF000);
            lpOut[2] = OutWord;
            lpIn += 4;
            lpOut += 3;
            NumPixel -= 8;
        }
    }
```

6.2.4 Writing of Packed Data as Image File Data

The packed data are written as 8-bit bytes. Within each 32-bit word packed with image data, the least significant 8-bits are sent first, and the most significant 8-bits are sent last; and the 32-bit words are sent in the order described in Section 6.2.1.

The figure below shows how 3 32-bit words of packed data in memory, produced by the algorithm given in pseudocode in Section 6.2.3.1 and representing 8 successive photosite values starting with a green photosite from a green-red row, would be written to a file as 12 successive 8-bit bytes.



Annex A Bibliography (Informative)

"ARRI Look Files in the ALEXA Camera" white paper, http://www.arri.com/camera/alexa/downloads/, 3 December 2013

ASC Color Decision List (ASC CDL) Transfer Functions and Interchange Syntax, ASC-CDL_Release1.2, Joshua Pines and David Reisner, 2009-05-04

DCI Digital Cinema System Specification v. 1.2 with Errata as of 30 August 2012 Incorporated, http://dcimovies.com/specification/DCI_DCSS_v12_with_errata_2012-1010.pdf