SER format description

Overview

Ser format consist of three parts:

- ❖ Header
  with fixed size of 178 Byte

- ❖ Image frame data
  with variable byte size of:
  <Pixeldepth (2Byte or 1Byte)> **x** <Image width> **x** <Image height> **x** <Total amount of Images>

- ❖ Trailer
  Optional. Byte size of 8 **x** <Total amount of Images>

Header

<u>1_FileID</u>

  Format: String
  Length: 14 Byte (14 characters)
  Content: "LUCAM-RECORDER"

<u>2_LuID</u>

  Format: Integer_32
  Length: 4 Byte
  Content: Lumenera camera series ID

<u>3_ColorID</u>

  Format: Integer_32
  Length: 4 Byte
  Content: MONO   = 0
                  BAYER_RGGB = 8
                  BAYER_GRBG = 9
                  BAYER_GBRG = 10
                  BAYER_BGGR = 11
                  BAYER_CYYM = 16
                  BAYER_YCMY = 17
                  BAYER_YMCY = 18
                  BAYER_MYYC = 19

4_LittleEndian

Format: Integer_32
Length: 4 Byte
Content: 0 for Big endian byte order in 16Bit pixel format
          1 for Little endian byte order in 16Bit pixel format

5_ImageWidth

Format: Integer_32
Length: 4 Byte
Content: Width of every image in pixel

6_ImageHeight

Format: Integer_32
Length: 4 Byte
Content: Height of every image in pixel

7_PixelDepth

Format: Integer_32
Length: 4 Byte
Content: True bit depth of an pixel
          If PixelDepth <= 8:   One pixel is stored in one Byte: BytePerPixel=1
          If PixelDepth > 8:   One pixel is stored in two Byte: BytePerPixel=2

8_FrameCount

Format: Integer_32
Length: 4 Byte
Content: Amount of image frames in SER file

9_Observer

Format: String
Length: 40 Byte (40 characters)
Content: Name of observer

10_Instrume

Format: String
Length: 40 Byte (40 characters)
Content: Name of used camera

11_Telescope

Format: String
Length: 40 Byte (40 characters)
Content: Name of used telescope

<u>12_DateTime</u>

Format: Date
Length: 8 Byte
Content: Start time of image stream
             ❖ If value = MinValue Then no Time data were stored
             ❖ If value = MinValue then SER file does not contain a Time stamp trailer

<u>13_DateTime_UTC</u>

Format: Date
Length: 8 Byte
Content: Start time of image stream in UTC

Image Data

Image data starts at File start offset decimal 178
Size of every image frame in byte is: 5_ImageWidth x 6_ImageHeigth x BytePerPixel

Trailer in detail

Trailer starts at byte offest: 8_FrameCount x 5_ImageWidth x 6_ImageHeigth x BytePerPixel

Trailer contains 8Byte time stamps for every image frame

<u>According to Microsoft documentation the used time stamp has the following format:</u>

"Holds IEEE 64-bit (8-byte) values that represent dates ranging from January 1 of the year 0001 through December 31 of the year 9999, and times from 12:00:00 AM (midnight) through 11:59:59.9999999 PM. Each increment represents 100 nanoseconds of elapsed time since the beginning of January 1 of the year 1 in the Gregorian calendar. The maximum value represents 100 nanoseconds before the beginning of January 1 of the year 10000."

According to the findings of Raoul Behrend, Université de Genève, the date record is not a 64 bits unsigned integer as stated, but a 62 bits unsigned integer. He got no information about the use of the two MSB.