

TIFF TAG Information Reader

TIFF (tagged image format specification) is a ubiquitous file format to store photometric information. As the acronym suggests the image is tagged with information. The Windows command shell utility tiffinfo.exe is a TIFF reader which can be downloaded from tiffinfo.zip. It has two basic modes - the primary displays tiff information tags while the secondary provides a means to view TIFF data as a HEX dump. Usage is:

```
>tiffinfo
usage: tiffinfo [-tvxX] File(s) [MoreFiles] -- for tiff information
        tiffinfo -b File [Start [Length [Width]]] - Hex Dump
            default dump start=0x0, length=0x100, width=0x10
        -? This help
        -b Dump file starting at DumpStart in 8-bit hex bytes
        -tTAG Dump TAG data
        -v Verbose, adds brief description of tag
        -x Give tiff info in Hex (default is Decimal)
        -X Give tiff info in both Hex and Decimal
Version 2.02
Wildtramper.com, Copyright (c) 2003-2009, All rights reserved
```

When executed in the primary form, the first line provides general TIFF file information, the second line defines the organization of subsequent TIFF tags, while the remaining lines are the file's TIFF tags. So to obtain TIFF information for a file IMG_8290.CR2:

```
>tiffinfo IMG_8290.CR2
IMG_8290.CR2: FileID=II, Ver=2A, IFD #1 w/ 14 Tags
TagName(ID,Type,Len,Ofst) =Value
...
where
1st line: IMG_8290.CR2: FileID=II, Ver=2A, IFD #1 w/ 14 Tags
FileID - II or MM for Intel or Motorola data formats
Ver     - TIFF version number
IFD     - Image file directory
2nd line: TagName(ID,Type,Len,Ofst) =Value
ID      - TAG ID in hexadecimal
Type    - Data type enumerator in hexadecimal
        1 - 8-bit byte
        2 - 8-bit ASCII code
        3 - 16-bit unsigned int
        4 - 32-bit unsigned int
        5 - 2 LONGs - numerator then denominator
        6 - 8-bit signed int
        7 - 8-bit contain anything
        8 - 16-bit signed int
        9 - 32-bit signed int
        a - 2 SLONGs - numerator then denominator
        b - 4-byte IEEE float
        c - 8-byte IEEE float
Len     - Number data of Type in hexadecimal
Ofst    - Offset from the start of file to the data entry in hexadecimal
Value   - Value(s) and sometime explanatory text information
```

Beware when viewing data at specified offsets that data may be organized in either LITTLE or BIG Endian. This is demarked by the FileID of either {II} or {MM} for Motorola or Intel formats.

As a more complete example, a Canon 5D CR2 image is encapsulated as a complex TIFF structure with multiple and sometimes multi-level IFDs. The dump of one such image contains 4 IFDs with an

embedded TIFF structure for its EXIF tag. When the TIFF information is displayed, IFD tags are indented to denote precedence. The dump of one of these files is:

```
>tiffinfo IMG_8290.CR2
IMG_8290.CR2: FileID=II, Ver=2A, IFD #1 w/ 14 Tags
  TagName(ID,Type,Len,Ofst)      =Value
  ImageWidth(0100,3,1,001A)      =2496
  ImageLength(0101,3,1,0026)     =1664
  BitsPerSample(0102,3,3,00BE)   =8 8 8
  Compression(0103,3,1,003E)     =6 {JpegCompression}
  Make(010F,2,6,00C4)            ="Canon"
  Model(0110,2,D,00CA)           ="Canon EOS 5D"
  StripOffsets(0111,4,1,0062)    =92361
  Orientation(0112,3,1,006E)     =1 {TopLeft, Normal}
  StripByteCounts(0117,4,1,007A)=2825048
  XResolution(011A,5,1,00EA)     =72/1 (=72)
  YResolution(011B,5,1,00F2)     =72/1 (=72)
  ResolutionUnit(0128,3,1,009E)  =2 {InchUnits}
  DateTime(0132,2,14,00FA)       ="2009:04:28 09:07:11"
  Exif(8769,4,1,00B6)           =270
    TagName(ID,Type,Len,Ofst)      =Value
    ExposureTime(829A,5,1,0264)    =1/125 (=0.008)
    FNumber(829D,5,1,026C)         =8/1 (=8)
    ExposureProgram(8822,3,1,0130)  =3
    ISOSpeedRatings(8827,3,1,013C)  =200
    ExifVersion(9000,7,4,0148)      =48 50 50 49
    DateTimeOriginal(9003,2,14,0274) ="2009:04:28 09:07:11"
    DateTimeDigitized(9004,2,14,0288) ="2009:04:28 09:07:11"
    ComponentsConfiguration(9101,7,4,016C) =1 2 3 0
    ShutterSpeedValue(9201,A,1,029C)  =458752/65536 (=7)
    ApertureValue(9202,5,1,02A4)     =393216/65536 (=6)
    ExposureBiasValue(9204,A,1,02AC)  =0/1 (=0)
    MeteringMode(9207,3,1,019C)       =0
    Flash(9209,3,1,01A8)             =16
    FocalLength(920A,5,1,02B4)        =96/1 (=96)
    MakerNote(927C,7,12548,02BC)     =29 0 1 0 3 0 46 0 0 0 30 4 0 0 ...
    UserComment(9286,7,108,12804)     =0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ...
    FlashpixVersion(A000,7,4,01D8)    =48 49 48 48
    ColorSpace(A001,3,1,01E4)         =1
    PixelXDimension(A002,3,1,01F0)    =4368
    PixelYDimension(A003,3,1,01FC)    =2912
    Interoperability(A005,4,1,0208)   =76044
    FocalPlaneXResolution(A20E,5,1,1292A) =4368000/1415 (=3086.93)
    FocalPlaneYResolution(A20F,5,1,12932) =2912000/942 (=3091.3)
    FocalPlaneResolutionUnit(A210,3,1,022C) =2 {InchUnits}
    CustomRendered(A401,3,1,0238)     =0
    ExposureMode(A402,3,1,0244)       =0
    WhiteBalance(A403,3,1,0250)       =0
    SceneCaptureType(A406,3,1,025C)   =0
IMG_8290.CR2: FileID=II, Ver=2A, IFD #2 w/ 2 Tags
  TagName(ID,Type,Len,Ofst)      =Value
  JPEGInterchangeFmt(0201,4,1,12944) =76348
  JPEGInterchangeFmtLen(0202,4,1,12950)=16013
IMG_8290.CR2: FileID=II, Ver=2A, IFD #3 w/ 11 Tags
  TagName(ID,Type,Len,Ofst)      =Value
  ImageWidth(0100,3,1,12962)      =384
  ImageLength(0101,3,1,1296E)     =256
  BitsPerSample(0102,3,3,129E2)   =8 8 8
  Compression(0103,3,1,12986)     =6 {JpegCompression}
  Photometric(0106,3,1,12992)     =2 {RGB}
  StripOffsets(0111,4,1,1299E)    =2917409
  SamplesPerPixel(0115,3,1,129AA)=3
```

```

RowsPerStrip(0116,3,1,129B6)    =256
StripByteCounts(0117,4,1,129C2)=294912
PlanarConfig(011C,3,1,129CE)    =1  {ChunkyFormat}
UndefinedTag(C5D9,4,1,129DA)    =2
IMG_8290.CR2: FileID=II, Ver=2A, IFD #4 w/ 6 Tags
TagName(ID,Type,Len,Ofst)       =Value
Compression(0103,3,1,129F2)     =6  {JpegCompression}
StripOffsets(0111,4,1,129FE)    =3212321
StripByteCounts(0117,4,1,12A0A)=12426836
UndefinedTag(C5D8,4,1,12A16)    =1
UndefinedTag(C5E0,4,1,12A22)    =1
Cr2Slice(C640,3,3,12A36)       =1 2238 2238

```

As a practical example to the dump data feature of tiffinfo.exe, the 4th IFD of this CR2 file contains the lossless JPEG compressed representation of the image. So if you wanted to dump this raw data, you would first query the StripOffsets (TAG #0x0111) value with hexadecimal output, that is:

```

>tiffinfo -Xt0111 IMG_8290.CR2
IMG_8290.CR2: FileID=II, Ver=2A, IFD #1 w/ 14 Tags
  TagName(ID,Type,Len,Ofst)  =Value
  StripOffsets(0111,4,1,0062)=92361 (000168C9)
0062: 000168C9
IMG_8290.CR2: FileID=II, Ver=2A, IFD #2 w/ 2 Tags
  TagName(ID,Type,Len,Ofst)  =Value
IMG_8290.CR2: FileID=II, Ver=2A, IFD #3 w/ 11 Tags
  TagName(ID,Type,Len,Ofst)  =Value
  StripOffsets(0111,4,1,1299E)=2917409 (002C8421)
1299E: 002C8421
IMG_8290.CR2: FileID=II, Ver=2A, IFD #4 w/ 6 Tags
  TagName(ID,Type,Len,Ofst)  =Value
  StripOffsets(0111,4,1,129FE)=3212321 (00310421)
129FE: 00310421

```

This yields a StripOffset from the 4th IFD of 0x00310421 bytes from the top of file. To view this data the second form of tiffinfo.exe is used. Thus:

```

>tiffinfo -b IMG_8290.CR2 310421
310421: FF D8 FF C4 00 3E 00 00 02 02 02 03 01 01 01 01 .....>.....
310431: 00 00 00 00 00 00 00 04 05 03 06 07 02 08 00 01 .....
310441: 09 0A 0C 0B 01 00 02 02 02 03 01 01 01 00 00 .....
310451: 00 00 00 00 00 04 05 03 06 07 02 08 00 01 09 0A .....
310461: 0C 0B FF C3 00 0E 0C 0B 8A 08 BE 02 01 11 00 02 .....
310471: 11 00 FF DA 00 0A 02 01 00 02 10 01 00 00 FF 00 .....
310481: 08 7F F0 85 F6 2D 37 F2 2F 66 DA 2D B6 D3 DD 3C .....-7./f.-...<
310491: EF 65 EF BD D7 6F 77 DB BA 47 D8 B6 EF 77 F3 5F .e...ow..G...w._
3104A1: 3B DD B7 DA 3E F9 E4 9A 77 6D 3D F7 DD 7C EE DD ;...>...wm=...|..
3104B1: F3 7D 35 F3 6D FD D7 5D BD D7 BE ED E7 BB 47 EF .}5.m...].....G.
3104C1: BA ED EC 5B F7 49 25 F2 3F 75 F3 BD F7 BD F3 7D ...[.I%.?u.....}
3104D1: 63 96 4D 7C F7 4D 3C 97 68 7B AE DB 45 EC 9D D7 c.M|.M<.h{..E...
3104E1: 6F 62 DE 7F 35 D3 C9 35 D2 4E FB DF 76 D7 5F 76 ob..5..5.N..v._v
3104F1: 8A 39 36 F7 68 BC DB 4E EF EE 9B 4B A6 9E F7 CD .96.h..N...K....
310501: 77 97 48 FC DB DD 3C DB 6E FB D8 E3 90 AD 46 DA w.H...<.n.....F.
310511: 58 F6 97 4D 3B AF BD F3 BA F7 59 26 EC 52 CE 3C X..M;.....Y&.R.<

```

This then displays the first few bytes of lossless JPEG compressed data. Note the JPEG markers {FFD8} for Start Of Image, {FFC4} for Define Huffman Table(s), {FFC3} for Start of Frame (Lossless/sequential decoding), and {FFDA} for Start of Scan. Of course these can be parsed to extract their meaning.

Good Luck.