NAME

JPEG Lossless compressor/decompressor – JPEG Lossless compressor/decompressor for compressed image sequences

DESCRIPTION

The Joint Photographic Experts Group (JPEG) is a joint ISO/CCITT technical committee. JPEG has developed a general-purpose international standard for the compression of continuous tone (grayscale or color) images. The standard has three categories:

The baseline specification,
An extended features specification, and
A lossless compression specification.

The baseline specification is a simple coding method based on the Discrete Cosine Transform (DCT) and uniform quantization in combination with statistical Huffman coding techniques for 8-bit image components. The extended features specification provides support for higher image precision, alternate statistical coding techniques, and progressive operation. The lossless compression technique is a 2-dimensional predictive differential coding technique, which also combines with statistical coding.

For more information and example programs, see the *XIL Programmer's Guide.*

The XIL implementation supports the baseline specification, "Jpeg" and the lossless specification, "JpegLL". Consult **Jpeg**(3) for more information on the XIL library's JPEG implementation.

Creating a CIS

To compress a compressed image sequence (CIS) with the XIL JPEG Lossless compressor, specify "JpegLL" for the *compressorname* argument in **xil_cis_create**(3).

Getting and Setting Attributes

Use **xil_cis_get_attribute**(3) and **xil_cis_set_attribute**(3) to get and set JPEG Lossless CIS attributes. These attributes are described in the following sections. Refer to the example section for additional information.

JpegLL Compression Attributes

The following paragraphs describe the JPEG Lossless CIS attributes available with the XIL library. All structures and enumerations are defined via **xil.h.** Note that some attributes are "set-only" and others are "get-only." This is noted under the *Access* heading for each attribute.

Note that if you are setting an attribute and that attribute is a structure, you must pass the address of that structure. If you are getting an attribute, you always pass its address.

COMPRESSED_DATA_FORMAT

Description	defines output format for JPEG Lossless compressor
Access	set
Type	typedef enum{ INTERCHANGE, ABBREVIATED_FORMAT } XilJpegCompressedDataFormat;

Values	*INTERCHANGE:* Use JPEG interchange format. All quantization and entropy-coding table specifications needed by the decoding process are included in each compressed frame. *ABBREVIATED_IMAGE:* Use JPEG abbreviated format for compressed images. Quantization and entropy-coding table specifications are not included in a compressed frame if the specifications are defined in a previous frame in the compressed sequence. If any table values change after they are defined in the compressed sequence, a new table definition is included in the first compressed frame that uses the new table values.
Default	*ABBREVIATED_IMAGE*
Notes	This does not include the third type: *ABBREVIATED_TABLE,* in which a frame contains only table specifications. However, the decoder will accept this format.

ENCODE_INTERLEAVED

Description	If the image to compress is composed of 4 bands or less, having this attribute set to TRUE will generate an interleaved JPEG-compliant bitstream (for example, 1:1:1:1 for a 4 banded image). If the number of bands exceeds 4 or if this attribute is set to FALSE, a noninterleaved JPEG-compliant bitstream is generated.
Access	set-only
Type	Xil_boolean
Values	*TRUE:* For images of 4 bands or less, produce an interleaved JPEG-compliant bitstream. *FALSE:* Produce a noninterleaved JPEG-compliant bitstream.
Default	*TRUE*

HUFFMAN_TABLE

Description	Set values in specified Huffman table
Access	set-only
Type	(see below)

```
typedef struct {
        int table;
        XilJpegHTableType type;
        XilJpegHTableValue *value;
} XilJpegHTable;

typedef enum {
        DC, AC
} XilJpegHTableType;
```

```
typedef struct {
        int bits;
        int pattern;
} XilJpegHTableValue;
```

Values *table:* For Lossless JPEG, you can have a value in the range from 0-3.

type: For Lossless JPEG, you must have value *DC,* and there are 17 entries in the value array.

value: A pointer to an array of dyads, where the first element 'bits' indicates the number of valid bits in the second element 'pattern'.

Default By default, the values in each of the tables are pre-initialized to the example values given in Annex K of the ANSI JPEG specification.

Default values for *table 0, type DC* are given in Table K.3 and are useful for the *DC* coefficients of the luminance band of 8-bit Y,Cb,Cr images. Default values for *table 1, type DC* are given in Table K.4 and are useful for the *DC* coefficients of the chrominance bands of 8-bit Y,Cb,Cr images.

Default values for *table 0, type AC* are given in Table K.5 and are useful for the *AC* coefficients of the luminance band of 8-bit Y,Cb,Cr images. Default values for *table 1, type AC* are given in Table K.6 and are useful for the *AC* coefficients of the chrominance bands of 8-bit Y,Cb,Cr images.

BAND_HUFFMAN_TABLE

Description Associates a band of an image to a Huffman table

Access set-only

Type

```
typedef struct {
        int band;
        int table;
        XilJpegHTableType type;
} XilJpegBandHTable;
```

Values *band:* Can have a value in the range 0-255.

table: For Lossless JPEG, you can have a value in the range 0-3.

type: For Lossless JPEG, you must have value *DC.*

Default — *band 0's DC* component is associated to *table 0, type DC* and *band 0's AC* component is associated to *table 0, type AC.* All other bands' *DC* component is associated to *table 1, type DC,* and their *AC* component is associated to *table 1, type AC.*

Notes — Bands may be associated to tables that have not yet been set. However, the tables must be set before a call to compress is made or an error occurs.

OPTIMIZE_HUFFMAN_TABLES

Description — Provide a hint to the compressor, enabling it to generate optimal Huffman tables instead of using the default example values specified in the ANSI specification. This is only a hint; the compressor is free to ignore the hint. For Lossless JPEG, setting this option attribute on or off keeps the current tables loaded. No optimal Huffman tables are provided for Lossless JPEG.

Access — set-only

Type — Xil_boolean

Values — *TRUE:* Huffman tables may vary from image to image to achieve higher compression.

FALSE: Use fixed Huffman tables for each image in the sequence.

Default — *FALSE*

LOSSLESS_BAND_SELECTOR

Description — Associates a band of an image to a predictor selection for the Lossless JPEG compressor. In the following discussion under the *Values* heading, Px = prediction for pixel "x", A = pixel left, B = pixel above, C = pixel diagonally above and left.

```
C  B
A  x
```

Access — set-only

Type —

```
typedef struct {
        int band;
        XilJpegLLBandSelectorType selector;
} XilJpegLLBandSelector;

typedef enum {
              ONE_D1,ONE_D2,ONE_D3,TWO_D1,TWO_D2,
                     TWO_D3,TWO_D4
 }  XilJpegLLBandSelectorType;
```

Values *band:* Can have a value in the range 0-255.

NO_PRED: Invalid selection for Lossless JPEG.

ONE_D1: Px = A

ONE_D2: Px = B

ONE_D3: Px = C

TWO_D1: Px = A + B - C

TWO_D2: Px = A + ((B - C)/2)

TWO_D3: Px = B + ((A - C)/2)

TWO_D4: Px = (A + B)/2

Default All bands default to selector ONE_D1.

LOSSLESS_BAND_PT_TRANSFORM

Description Associates a band of an image with a point transform, PtTransform, for the Lossless JPEG compressor. If PtTransform is non-zero, the input image band is divided by 2**PtTransform before lossless encoding.

Access set-only

Type
```
typedef struct {
        int band;
        int PtTransform;
} XilJpegLLBandPtTransform;
```

Values *band:* Can have a value in the range 0-255.

PtTransform: Can have a value in the range 0-15.

Default All bands default to PtTransform = 0.

EXAMPLES The following example opens and closes a JPEG Lossless CIS using the XIL library:

```
XilSystemState State;
XilCis cis;
State = xil_open();
cis = xil_cis_create(State, "JpegLL");

-- calls to JpegLL-specific compression routines --

xil_cis_destroy(cis);
xil_close(State);
```

The following example sets a JPEG Lossless CIS attribute called *ENCODE_INTERLEAVED* to TRUE. Note that because this attribute is not a structure, it is not necessary to pass the address of *ENCODE_INTERLEAVED* when setting it.

```
XilCis cis;

xil_cis_set_attribute(cis,"ENCODE_INTERLEAVED", (void *) TRUE);
```

The following example returns the value of a JPEG Lossless CIS attribute called *ENCODE_INTERLEAVED.* Note that when getting an attribute it is always necessary to pass the address.

```
Xil_boolean encode_type;
XilCis cis;

xil_cis_get_attribute(cis, "ENCODE_INTERLEAVED", (void **) &encode_type);
```

NOTES

The **xil_cis_set_attribute** () and **xil_cis_get_attribute** () calls are used to modify the default behavior of a specific compressor. Generic attributes of compressors are set by individual function calls.

SEE ALSO

xil_cis_create(3), **xil_cis_get_attribute**(3), **xil_cis_get_bits_ptr**(3), **xil_compress**(3), **xil_decompress**(3).