

JasPer  
2.0.33

Generated by Doxygen 1.9.8



<b>1 Reference Manual (Version 2.0.33)</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 News . . . . .	1
1.3 License . . . . .	4
1.4 Reporting Bugs . . . . .	5
<b>2 Getting Started</b>	<b>7</b>
2.1 Installation . . . . .	7
2.2 Organization of the Manual . . . . .	9
<b>3 Frequently Asked Questions (FAQ)</b>	<b>11</b>
<b>4 Known Bugs</b>	<b>13</b>
<b>5 Todo List</b>	<b>15</b>
<b>6 Topic Index</b>	<b>17</b>
6.1 Topics . . . . .	17
<b>7 Class Index</b>	<b>19</b>
7.1 Class List . . . . .	19
<b>8 File Index</b>	<b>21</b>
8.1 File List . . . . .	21
<b>9 Topic Documentation</b>	<b>23</b>
9.1 Initialization and Clean Up . . . . .	23
9.1.1 Detailed Description . . . . .	23
9.1.2 Function Documentation . . . . .	23
9.1.2.1 jas_cleanup() . . . . .	23
9.1.2.2 jas_init() . . . . .	24
9.2 I/O Streams . . . . .	24
9.2.1 Detailed Description . . . . .	26
9.2.2 Macro Definition Documentation . . . . .	26
9.2.2.1 jas_stream_clearerr . . . . .	26
9.2.2.2 jas_stream_eof . . . . .	26
9.2.2.3 jas_stream_error . . . . .	27
9.2.2.4 jas_stream_getc . . . . .	27
9.2.2.5 jas_stream_getrwcoun . . . . .	27
9.2.2.6 jas_stream_getrwl . . . . .	28
9.2.2.7 jas_stream_peekc . . . . .	28
9.2.2.8 jas_stream_putc . . . . .	28

9.2.3 Function Documentation	29
9.2.3.1 jas_stream_close()	29
9.2.3.2 jas_stream_copy()	29
9.2.3.3 jas_stream_display()	30
9.2.3.4 jas_stream_fdopen()	30
9.2.3.5 jas_stream_flush()	30
9.2.3.6 jas_stream_fopen()	31
9.2.3.7 jas_stream_freopen()	31
9.2.3.8 jas_stream_gets()	32
9.2.3.9 jas_stream_gobble()	32
9.2.3.10 jas_stream_isseekable()	33
9.2.3.11 jas_stream_length()	33
9.2.3.12 jas_stream_memopen()	34
9.2.3.13 jas_stream_memopen2()	34
9.2.3.14 jas_stream_pad()	34
9.2.3.15 jas_stream_peek()	35
9.2.3.16 jas_stream_printf()	36
9.2.3.17 jas_stream_puts()	36
9.2.3.18 jas_stream_read()	36
9.2.3.19 jas_stream_rewind()	37
9.2.3.20 jas_stream_seek()	38
9.2.3.21 jas_stream_setrwcounr()	38
9.2.3.22 jas_stream_setrwlmit()	39
9.2.3.23 jas_stream_tell()	39
9.2.3.24 jas_stream_tmpfile()	39
9.2.3.25 jas_stream_ungetc()	40
9.2.3.26 jas_stream_write()	40
9.3 Image	41
9.3.1 Detailed Description	44
9.3.2 Macro Definition Documentation	44
9.3.2.1 jas_image_brx	44
9.3.2.2 jas_image_bry	44
9.3.2.3 jas_image_clrspc	44
9.3.2.4 jas_image_cmprof	45
9.3.2.5 jas_image_cmptbrx	45
9.3.2.6 jas_image_cmptbry	45
9.3.2.7 jas_image_cmptheight	45
9.3.2.8 jas_image_cmptthstep	45
9.3.2.9 jas_image_cmptprec	46

9.3.2.10	<a href="#">jas_image_cmptsgnd</a>	46
9.3.2.11	<a href="#">jas_image_cmpttlx</a>	46
9.3.2.12	<a href="#">jas_image_cmpttly</a>	46
9.3.2.13	<a href="#">jas_image_cmptvstep</a>	46
9.3.2.14	<a href="#">jas_image_cmptwidth</a>	46
9.3.2.15	<a href="#">jas_image_height</a>	47
9.3.2.16	<a href="#">JAS_IMAGE_MAXFMTS</a>	47
9.3.2.17	<a href="#">jas_image_numcmpts</a>	47
9.3.2.18	<a href="#">jas_image_setclrspc</a>	47
9.3.2.19	<a href="#">jas_image_setcmprof</a>	47
9.3.2.20	<a href="#">jas_image_tlx</a>	47
9.3.2.21	<a href="#">jas_image_tly</a>	48
9.3.2.22	<a href="#">jas_image_width</a>	48
9.3.3	<a href="#">Typedef Documentation</a>	48
9.3.3.1	<a href="#">jas_image_cmpttype_t</a>	48
9.3.3.2	<a href="#">jas_image_colorspc_t</a>	48
9.3.3.3	<a href="#">jas_image_coord_t</a>	48
9.3.3.4	<a href="#">jas_image_smpltype_t</a>	48
9.3.4	<a href="#">Function Documentation</a>	49
9.3.4.1	<a href="#">jas_image_addcmpt()</a>	49
9.3.4.2	<a href="#">jas_image_addfmt()</a>	49
9.3.4.3	<a href="#">jas_image_chclrspc()</a>	49
9.3.4.4	<a href="#">jas_image_clearfmts()</a>	49
9.3.4.5	<a href="#">jas_image_cmpt_domains_same()</a>	49
9.3.4.6	<a href="#">jas_image_copy()</a>	50
9.3.4.7	<a href="#">jas_image_copycmpt()</a>	50
9.3.4.8	<a href="#">jas_image_create()</a>	50
9.3.4.9	<a href="#">jas_image_create0()</a>	50
9.3.4.10	<a href="#">jas_image_decode()</a>	50
9.3.4.11	<a href="#">jas_image_delcmpt()</a>	51
9.3.4.12	<a href="#">jas_image_depalettize()</a>	51
9.3.4.13	<a href="#">jas_image_destroy()</a>	51
9.3.4.14	<a href="#">jas_image_dump()</a>	51
9.3.4.15	<a href="#">jas_image_encode()</a>	51
9.3.4.16	<a href="#">jas_image_fmtfromname()</a>	52
9.3.4.17	<a href="#">jas_image_fmtostr()</a>	52
9.3.4.18	<a href="#">jas_image_getcmptbytype()</a>	52
9.3.4.19	<a href="#">jas_image_getfmt()</a>	52
9.3.4.20	<a href="#">jas_image_ishomosamp()</a>	52

9.3.4.21 <a href="#">jas_image_lookupfmtbyid()</a>	52
9.3.4.22 <a href="#">jas_image_lookupfmtbyname()</a>	53
9.3.4.23 <a href="#">jas_image_rawsize()</a>	53
9.3.4.24 <a href="#">jas_image_readcmpt()</a>	53
9.3.4.25 <a href="#">jas_image_readcmpt2()</a>	53
9.3.4.26 <a href="#">jas_image_readcmptsample()</a>	53
9.3.4.27 <a href="#">jas_image_sampcmpt()</a>	54
9.3.4.28 <a href="#">jas_image_strtofmt()</a>	54
9.3.4.29 <a href="#">jas_image_writecmpt()</a>	54
9.3.4.30 <a href="#">jas_image_writecmpt2()</a>	54
9.3.4.31 <a href="#">jas_image_writecmptsample()</a>	55
<b>10 Class Documentation</b>	<b>57</b>
10.1 <a href="#">jas_image_cmpt_t Struct Reference</a>	57
10.1.1 Detailed Description	57
10.2 <a href="#">jas_image_cmptparm_t Struct Reference</a>	57
10.2.1 Detailed Description	57
10.3 <a href="#">jas_image_fmtinfo_t Struct Reference</a>	58
10.3.1 Detailed Description	58
10.4 <a href="#">jas_image_fmtops_t Struct Reference</a>	58
10.4.1 Detailed Description	58
10.5 <a href="#">jas_image_t Struct Reference</a>	58
10.5.1 Detailed Description	58
<b>11 File Documentation</b>	<b>59</b>
11.1 <a href="#">bmp_cod.h</a>	59
11.2 <a href="#">bmp_enc.h</a>	62
11.3 <a href="#">jas_cm.h File Reference</a>	63
11.3.1 Detailed Description	63
11.4 <a href="#">jas_cm.h</a>	63
11.5 <a href="#">jas_compiler.h File Reference</a>	66
11.5.1 Detailed Description	66
11.6 <a href="#">jas_compiler.h</a>	67
11.7 <a href="#">jas_debug.h File Reference</a>	68
11.7.1 Detailed Description	68
11.8 <a href="#">jas_debug.h</a>	68
11.9 <a href="#">jas_dll.h File Reference</a>	70
11.9.1 Detailed Description	70
11.10 <a href="#">jas_dll.h</a>	70
11.11 <a href="#">jas_fix.h File Reference</a>	71

---

11.11.1 Detailed Description . . . . .	72
11.12 jas_fix.h . . . . .	72
11.13 jas_getopt.h File Reference . . . . .	76
11.13.1 Detailed Description . . . . .	76
11.14 jas_getopt.h . . . . .	76
11.15 jas_icc.h File Reference . . . . .	78
11.15.1 Detailed Description . . . . .	78
11.16 jas_icc.h . . . . .	78
11.17 jas_image.h File Reference . . . . .	84
11.17.1 Detailed Description . . . . .	87
11.18 jas_image.h . . . . .	87
11.19 jas_init.h File Reference . . . . .	94
11.19.1 Detailed Description . . . . .	94
11.20 jas_init.h . . . . .	95
11.21 jas_malloc.h File Reference . . . . .	96
11.21.1 Detailed Description . . . . .	96
11.22 jas_malloc.h . . . . .	96
11.23 jas_math.h File Reference . . . . .	98
11.23.1 Detailed Description . . . . .	98
11.24 jas_math.h . . . . .	98
11.25 jas_seq.h File Reference . . . . .	103
11.25.1 Detailed Description . . . . .	103
11.26 jas_seq.h . . . . .	103
11.27 jas_stream.h File Reference . . . . .	109
11.27.1 Detailed Description . . . . .	111
11.28 jas_stream.h . . . . .	111
11.29 jas_string.h File Reference . . . . .	117
11.29.1 Detailed Description . . . . .	117
11.30 jas_string.h . . . . .	117
11.31 jas_tmr.h File Reference . . . . .	118
11.31.1 Detailed Description . . . . .	119
11.32 jas_tmr.h . . . . .	119
11.33 jas_tvp.h File Reference . . . . .	120
11.33.1 Detailed Description . . . . .	120
11.34 jas_tvp.h . . . . .	121
11.35 jas_types.h File Reference . . . . .	123
11.35.1 Detailed Description . . . . .	123
11.36 jas_types.h . . . . .	123
11.37 jas_version.h File Reference . . . . .	125

---

11.37.1 Detailed Description . . . . .	125
11.38 jas_version.h . . . . .	125
11.39 jasper.h File Reference . . . . .	127
11.39.1 Detailed Description . . . . .	127
11.40 jasper.h . . . . .	127
11.41 jp2_cod.h . . . . .	128
11.42 jp2_dec.h . . . . .	132
11.43 jpc_bs.h . . . . .	134
11.44 jpc_cod.h . . . . .	137
11.45 jpc_cs.h . . . . .	138
11.46 jpc_dec.h . . . . .	148
11.47 jpc_enc.h . . . . .	156
11.48 jpc_fix.h . . . . .	164
11.49 jpcflt.h . . . . .	167
11.50 jpc_math.h . . . . .	168
11.51 jpc_mct.h . . . . .	170
11.52 jpc_mqcod.h . . . . .	171
11.53 jpc_mqdec.h . . . . .	173
11.54 jpc_mqenc.h . . . . .	176
11.55 jpc_qmfb.h . . . . .	178
11.56 jpc_t1cod.h . . . . .	180
11.57 jpc_t1dec.h . . . . .	184
11.58 jpc_t1enc.h . . . . .	185
11.59 jpc_t2cod.h . . . . .	186
11.60 jpc_t2dec.h . . . . .	190
11.61 jpc_t2enc.h . . . . .	191
11.62 jpc_tagtree.h . . . . .	193
11.63 jpc_tsfb.h . . . . .	195
11.64 jpc_util.h . . . . .	197
11.65 jpg_cod.h . . . . .	198
11.66 jpg_enc.h . . . . .	199
11.67 jpg_jpeglib.h . . . . .	200
11.68 mif_cod.h . . . . .	201
11.69 pgx_cod.h . . . . .	202
11.70 pgx_enc.h . . . . .	204
11.71 pnm_cod.h . . . . .	205
11.72 pnm_enc.h . . . . .	207
11.73 ras_cod.h . . . . .	208
11.74 ras_enc.h . . . . .	210



# Chapter 1

## Reference Manual (Version 2.0.33)

### 1.1 Introduction

JasPer is a collection of software (i.e., a library and application programs) for the coding and manipulation of images. This software can handle image data in a variety of formats. One such format supported by JasPer is the JPEG-2000 format defined in ISO/IEC 15444-1. This software was developed by [Michael Adams](#) from the Department of Electrical and Computer Engineering at the University of Victoria, Victoria, BC, Canada.

### 1.2 News

2.0.33 (2021-08-01)  
=====

- \* Fix a JP2/JPC decoder bug.
- \* Fix a build issue impacting some platforms.

2.0.32 (2021-04-18)  
=====

- \* Test release performed with GitHub Actions.

2.0.29 (2021-04-16)  
=====

- \* Loosen some overly tight restrictions on JP2 codestreams, which caused some valid codestreams to be rejected. (#289)

2.0.28 (2021-03-29)  
=====

- \* Fix potential null pointer dereference in the JP2/JPC decoder. (#269)
- \* Fix ignoring of JAS\_STREAM\_FILEOBJ\_NOCLOSE at stream close time. (#286)
- \* Fix integral type sizing problem in JP2 codec. (#284)

2.0.27 (2021-03-18)  
=====

- \* Check for an image containing no samples in the PGX decoder. (#271, #272, #273, #274, #275, #276, #281)
- \* Check for dimensions of zero in the JPC and JPEG decoders.

- \* Fix an arguably incorrect type for an integer literal in the PGX decoder. (#270)
- \* Check for an invalid component reference in the JP2 decoder. (#269)
- \* Check on integer size in JP2 decoder. (#278)

2.0.26 (2021-03-05)

=====

- \* Fix JP2 decoder bug that can cause a null pointer dereference for some invalid CDEF boxes. (#268) (CVE-2021-3467)

2.0.25 (2021-02-07)

=====

- \* Fix memory-related bugs in the JPEG-2000 codec resulting from attempting to decode invalid code streams. (#264, #265)  
This fix is associated with CVE-2021-26926 and CVE-2021-26927.
- \* Fix wrong return value under some compilers (#260)
- \* Fix CVE-2021-3272 heap buffer overflow in jp2\_decode (#259)

2.0.24 (2021-01-03)

=====

- \* Add JAS\_VERSION\_MAJOR, JAS\_VERSION\_MINOR, JAS\_VERSION\_PATCH for easier access to the Jasper version.
- \* Fixes stack overflow bug on Windows, where variable-length arrays are not available. (#256)

2.0.23 (2020-12-08)

=====

- \* Fix CVE-2020-27828, heap-overflow in cp\_create() in jpc\_enc.c  
<https://github.com/jasper-software/jasper/issues/252>

2.0.22 (2020-10-05)

=====

- \* Update manual
- \* Remove JPEG dummy codec. Jasper needs libjpeg for JPEG support
- \* Fix test suite build failure regarding disabled MIF codec (#249)
- \* Fix OpenGL/glut detection (#247)

2.0.21 (2020-09-20)

=====

- \* Fix ZDI-15-529  
<https://github.com/jasper-software/jasper/pull/245>
- \* Fix CVE-2018-19541 in decoder  
<https://github.com/jasper-software/jasper/pull/244>

2.0.20 (2020-09-05)

=====

- \* Fix several ISO/IEC 15444-4 conformance bugs
- \* Fix new variant of CVE-2016-9398
- \* Disable the MIF codec by default for security reasons (but it is still included in the library);  
in a future release, the MIF codec may also be excluded from the library by default
- \* Add documentation for the I/O streams library API

2.0.19 (2020-07-11)

=====

- \* Fix CVE-2018-9154  
<https://github.com/jasper-software/jasper/issues/215>  
<https://github.com/jasper-software/jasper/issues/166>  
<https://github.com/jasper-software/jasper/issues/175>  
<https://github.com/jasper-maint/jasper/issues/8>
- \* Fix CVE-2018-19541 in encoder  
<https://github.com/jasper-software/jasper/pull/199>  
<https://github.com/jasper-maint/jasper/issues/6>
- \* Fix CVE-2016-9399, CVE-2017-13751  
<https://github.com/jasper-maint/jasper/issues/1>
- \* Fix CVE-2018-19540  
<https://github.com/jasper-software/jasper/issues/182>  
<https://github.com/jasper-maint/jasper/issues/22>
- \* Fix CVE-2018-9055  
<https://github.com/jasper-maint/jasper/issues/9>
- \* Fix CVE-2017-13748  
<https://github.com/jasper-software/jasper/issues/168>
- \* Fix CVE-2017-5503, CVE-2017-5504, CVE-2017-5505  
<https://github.com/jasper-maint/jasper/issues/3>  
<https://github.com/jasper-maint/jasper/issues/4>  
<https://github.com/jasper-maint/jasper/issues/5>  
<https://github.com/jasper-software/jasper/issues/88>  
<https://github.com/jasper-software/jasper/issues/89>  
<https://github.com/jasper-software/jasper/issues/90>
- \* Fix CVE-2018-9252  
<https://github.com/jasper-maint/jasper/issues/16>
- \* Fix CVE-2018-19139  
<https://github.com/jasper-maint/jasper/issues/14>
- \* Fix CVE-2018-19543, CVE-2017-9782  
<https://github.com/jasper-maint/jasper/issues/13>  
<https://github.com/jasper-maint/jasper/issues/18>  
<https://github.com/jasper-software/jasper/issues/140>  
<https://github.com/jasper-software/jasper/issues/182>
- \* Fix CVE-2018-20570  
<https://github.com/jasper-maint/jasper/issues/11>  
<https://github.com/jasper-software/jasper/issues/191>
- \* Fix CVE-2018-20622  
<https://github.com/jasper-maint/jasper/issues/12>  
<https://github.com/jasper-software/jasper/issues/193>
- \* Fix CVE-2016-9398  
<https://github.com/jasper-maint/jasper/issues/10>
- \* Fix CVE-2017-14132  
<https://github.com/jasper-maint/jasper/issues/17>
- \* Fix CVE-2017-5499  
<https://github.com/jasper-maint/jasper/issues/2>  
<https://github.com/jasper-software/jasper/issues/63>
- \* Fix CVE-2018-18873  
<https://github.com/jasper-maint/jasper/issues/15>  
<https://github.com/jasper-software/jasper/issues/184>

- \* Fix <https://github.com/jasper-software/jasper/issues/207>
- \* Fix <https://github.com/jasper-software/jasper/issues/194> part 1
- \* Fix CVE-2017-13750
  - <https://github.com/jasper-software/jasper/issues/165>
  - <https://github.com/jasper-software/jasper/issues/174>
- \* New option `-DJAS_ENABLE_HIDDEN=true` to not export internal symbols in the public symbol table
- \* Fix various memory leaks
- \* Plenty of code cleanups, and performance improvements
- \* Some macros were changed to inline functions. This has the potential to impact some code that made assumptions about the implementation. Some potentially impacted macros include:
  - `jas_matrix_numrows`, `jas_matrix_numcols`
  - `jas_matrix_get`
  - `jas_seq_get`, `jas_seq_start`, `jas_seq_end`
  - `jpc_ms_gettype`
  - `jas_matrix_set` and `jas_seq_set` is affected differently; the old macro was an actual expression returning the value, while the new function does not.
 The following macros have been changed, too, but are likely not affected, since they have been an rvalue-expression anyway:
  - `JP2_DTYPETOBCP`, `JP2_BPCTODTYPE`
  - `JAS_IMAGE_CDT_{SETSGND,GETSGND,SETPREC,GETPREC}`
  - `jas_image_cmptdtype`
  - macros from here
  - `jas_matrix_setv`, `jas_matrix_getvref`
  - `jas_matrix_bind{row,col}`
  - the `jpc_fix_` family
  - the `JPC_QCX` and `JPC_COX` families

## 1.3 License

JasPer License Version 2.0

Copyright (c) 2001-2016 Michael David Adams  
 Copyright (c) 1999-2000 Image Power, Inc.  
 Copyright (c) 1999-2000 The University of British Columbia

All rights reserved.

Permission is hereby granted, free of charge, to any person (the "User") obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

1. The above copyright notices and this permission notice (which includes the disclaimer below) shall be included in all copies or substantial portions of the Software.
2. The name of a copyright holder shall not be used to endorse or promote products derived from the Software without specific prior written permission.

THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS

"AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY. EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES, AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH, PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.

## 1.4 Reporting Bugs

All bug reports should be submitted via the issue-tracking system provided by GitHub. To submit a bug report, go the the following URL and click on the "New issue" button:

<https://github.com/jasper-software/jasper/issues>

Please do not submit bug reports directly to the author of JasPer via email, as bug reports that are not submitted via the above issue-tracking system on GitHub are easy to be lost.



## Chapter 2

# Getting Started

The following sections are useful for getting started with the Jasper software:

- [Installation](#). Describes how to install the Jasper software.
- [Organization of the Manual](#). Briefly explains the organization of the manual (e.g., where to find things).

## 2.1 Installation

Installation  
=====

The process required to install Jasper is described below.

Installation on Systems Running Unix  
-----

In what follows, let `$SOURCE_DIR` denote the top-level directory of the Jasper software source tree (i.e., the directory containing the files named `LICENSE` and `INSTALL`) and let `$INSTALL_DIR` denote the target directory for installation.

1) Select an empty directory to use for building the software. Let `$BUILD_DIR` denote this directory.

2) Generate the makefiles used for building the software. To do this, invoke the command:

```
cmake -G "Unix Makefiles" -H$SOURCE_DIR -B$BUILD_DIR \  
-DCMAKE_INSTALL_PREFIX=$INSTALL_DIR $OPTIONS
```

where `$OPTIONS` corresponds to zero or more `-D` options as described below under the heading "Cmake Options".

3) Change the working directory to the build directory. To do this, use the command:

```
cd $BUILD_DIR
```

4) Build the code using the make utility. To do this, invoke the command:

```
make clean all
```

5) Run the test suite to ensure that the software seems to be working correctly. To do this, invoke the command:

```
make test
```

If more verbose output from the testing process is desired (e.g., to assist in diagnosing a problem), instead use the command:

```
make test ARGS="-V"
```

6) Install the software. To do this, invoke the command:

```
make install
```

#### Additional Remarks:

When building the Jasper software under Mac OSX, only the use of the native framework for OpenGL is officially supported. If the Freeglut library is installed on your system, you will need to ensure that the native GLUT library (as opposed to the Freeglut library) is used by the build process. This can be accomplished by adding an extra option to the cmake command line that resembles the following:

```
-DGLUT_glut_LIBRARY=/System/Library/Frameworks/GLUT.framework
```

#### Installation on Systems Running Microsoft Windows

In what follows, let %SOURCE\_DIR% denote the top-level directory of the Jasper software source tree (i.e., the directory containing the files named LICENSE and INSTALL) and let %INSTALL\_DIR% denote the target directory for installation.

1) Select an empty directory to use for building the software. Let %BUILD\_DIR% denote this directory.

2) Generate the project file needed to build the software with Microsoft Visual Studio. To do this, invoke the command:

```
cmake -G "Visual Studio 12 2013 Win64" -H%SOURCE_DIR% -B%BUILD_DIR% ^
-DCMAKE_INSTALL_PREFIX=%INSTALL_DIR% %OPTIONS%
```

where %OPTIONS% corresponds to zero or more -D options as described below under the heading "Cmake Options". (Note the caret symbol "^" above denotes line continuation.)

3) Build and install the software. To do this, invoke the command:

```
msbuild %build_dir%\INSTALL.vcxproj
```

#### Cmake Options

The option OPTION can be set to the value VALUE with a command-line option of the form -DOPTION=VALUE  
The following options are supported:

CMAKE\_INSTALL\_PREFIX  
Specify the installation directory.  
Value: A directory name.

CMAKE\_BUILD\_TYPE  
Specify the build type (i.e., release or debug).  
Valid values: Debug or Release

JAS\_ENABLE\_DOC



Enable the building of the documentation (which requires LaTeX).  
Valid values: true and false

JAS\_ENABLE\_LIBJPEG  
Enable the use of the JPEG library  
Valid values: true and false

JAS\_ENABLE\_OPENGL  
Enable the use of the OpenGL and GLUT libraries.  
Valid values: true and false

JAS\_ENABLE\_SHARED  
Enable the building of shared libraries.  
Valid values: true or false

JAS\_ENABLE\_HIDDEN  
Hide internal symbols? Enabling this results in a smaller binary.  
Valid values: true or false

JAS\_ENABLE\_32BIT  
Force the use of 32 bit integers? On 64 bit CPUs, JasPer historically used 64 bit integers which consumes more memory, is slower and has no advantages. This produces a different ABI, so the resulting library is not compatible with other builds.  
Valid values: true or false

JAS\_ENABLE\_ASAN  
Enable the Address Sanitizer.  
Valid values: true or false

JAS\_ENABLE\_USAN  
Enable the Undefined-Behavior Sanitizer.  
Valid values: true or false

JAS\_ENABLE\_LSAN  
Enable the Leak Sanitizer.  
Valid values: true or false

JAS\_ENABLE\_MSAN  
Enable the Memory Sanitizer.  
Valid values: true or false

## 2.2 Organization of the Manual

The library is partitioned into groups of related code called modules. The documentation is also partitioned in this way. The documentation for each of the various modules can be found in the [modules page](#).



## Chapter 3

# Frequently Asked Questions (FAQ)

The following is a list of common questions/problems encountered when using the library.

-



## Chapter 4

# Known Bugs

All bugs reported in Jasper are tracked using the issue-tracking functionality provided by GitHub. If you encounter a problem with Jasper and you would like to know if it is a known problem, please check the issue tracker for Jasper on GitHub, which can be found at the following URL:

`https://github.com/jasper-software/jasper/issues`

If you happen to find a bug that has not been previously reported, please report it so that it can be fixed. New bugs can be reported by creating a new issue using the page at the above URL.



# Chapter 5

## Todo List

Member `jas_stream_clearerr` (`stream`)

TODO/FIXME: Should this macro evaluate to void?

Member `jas_stream_copy` (`jas_stream_t *destination`, `jas_stream_t *source`, `int count`)

TODO/FIXME: count should probably be a `size_t`; return type `ssize_t`?

Member `jas_stream_display` (`jas_stream_t *stream`, `FILE *fp`, `int count`)

TODO/FIXME: should count be unsigned int or `size_t` instead of int?

Member `jas_stream_pad` (`jas_stream_t *stream`, `int count`, `int value`)

TODO: should the count be `size_t`; return type maybe `size_t`?

Member `jas_stream_printf` (`jas_stream_t *stream`, `const char *format`,...)

I think that the return type of int is okay here. It is consistent with printf and friends.

Member `jas_stream_setrwcoun`t (`jas_stream_t *stream`, `long rw_count`)

TODO/FIXME: Should this macro evaluate to void?





# Chapter 6

## Topic Index

### 6.1 Topics

Here is a list of all topics with brief descriptions:

Initialization and Clean Up . . . . .	23
I/O Streams . . . . .	24
Image . . . . .	41



# Chapter 7

## Class Index

### 7.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">jas_image_cmpt_t</a>	Image component class . . . . .	57
<a href="#">jas_image_cmptparm_t</a>	Component parameters class . . . . .	57
<a href="#">jas_image_fmtnfo_t</a>	Image format information . . . . .	58
<a href="#">jas_image_fmtnops_t</a>	Image format-dependent operations . . . . .	58
<a href="#">jas_image_t</a>	Image class . . . . .	58



# Chapter 8

## File Index

### 8.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">bmp_cod.h</a>	59
<a href="#">bmp_enc.h</a>	62
<a href="#">jas_cm.h</a>	
JasPer Color Management	63
<a href="#">jas_compiler.h</a>	
Compiler-related macros	66
<a href="#">jas_debug.h</a>	
JasPer Debugging-Related Functionality	68
<a href="#">jas_dll.h</a>	
Shared Library Macros	70
<a href="#">jas_fix.h</a>	
JasPer Fixed-Point Number Class	71
<a href="#">jas_getopt.h</a>	
Command Line Option Parsing Code	76
<a href="#">jas_icc.h</a>	
ICC Profile	78
<a href="#">jas_image.h</a>	
JasPer Image Class	84
<a href="#">jas_init.h</a>	
JasPer Initialization/Cleanup Code	94
<a href="#">jas_malloc.h</a>	
JasPer Memory Allocator	96
<a href="#">jas_math.h</a>	
Math-Related Code	98
<a href="#">jas_seq.h</a>	
Sequence/Matrix Library	103
<a href="#">jas_stream.h</a>	
I/O Stream Class	109
<a href="#">jas_string.h</a>	
String Library	117
<a href="#">jas_tmr.h</a>	
Timer Code	118

<a href="#">jas_tvp.h</a>	
Tag/Value Pair Parser . . . . .	120
<a href="#">jas_types.h</a>	
Primitive Types . . . . .	123
<a href="#">jas_version.h</a>	
JasPer Version . . . . .	125
<a href="#">jasper.h</a>	
JasPer Main Header . . . . .	127
<a href="#">jp2_cod.h</a>	128
<a href="#">jp2_dec.h</a>	132
<a href="#">jpc_bs.h</a>	134
<a href="#">jpc_cod.h</a>	137
<a href="#">jpc_cs.h</a>	138
<a href="#">jpc_dec.h</a>	148
<a href="#">jpc_enc.h</a>	156
<a href="#">jpc_fix.h</a>	164
<a href="#">jpc_ftt.h</a>	167
<a href="#">jpc_math.h</a>	168
<a href="#">jpc_mct.h</a>	170
<a href="#">jpc_mqcod.h</a>	171
<a href="#">jpc_mqdec.h</a>	173
<a href="#">jpc_mqenc.h</a>	176
<a href="#">jpc_qmfb.h</a>	178
<a href="#">jpc_t1cod.h</a>	180
<a href="#">jpc_t1dec.h</a>	184
<a href="#">jpc_t1enc.h</a>	185
<a href="#">jpc_t2cod.h</a>	186
<a href="#">jpc_t2dec.h</a>	190
<a href="#">jpc_t2enc.h</a>	191
<a href="#">jpc_tagtree.h</a>	193
<a href="#">jpc_tsfb.h</a>	195
<a href="#">jpc_util.h</a>	197
<a href="#">jpg_cod.h</a>	198
<a href="#">jpg_enc.h</a>	199
<a href="#">jpg_jpeglib.h</a>	200
<a href="#">mif_cod.h</a>	201
<a href="#">pgx_cod.h</a>	202
<a href="#">pgx_enc.h</a>	204
<a href="#">pnm_cod.h</a>	205
<a href="#">pnm_enc.h</a>	207
<a href="#">ras_cod.h</a>	208
<a href="#">ras_enc.h</a>	210

# Chapter 9

## Topic Documentation

### 9.1 Initialization and Clean Up

Initialization and Clean Up.

#### Functions

- JAS\_DLLEXPORT int [jas\\_init](#) (void)  
*Initialize the JasPer library.*
- JAS\_DLLEXPORT void [jas\\_cleanup](#) (void)  
*Perform any clean up for the JasPer library.*

#### 9.1.1 Detailed Description

Initialization and Clean Up.

#### 9.1.2 Function Documentation

##### 9.1.2.1 [jas\\_cleanup\(\)](#)

```
JAS_DLLEXPORT void jas_cleanup (  
    void )
```

Perform any clean up for the JasPer library.

This function performs any clean up for the JasPer library.

### 9.1.2.2 jas\_init()

```
JAS_DLLEXPORT int jas_init (
    void )
```

Initialize the JasPer library.

This function must be called before any other code in the JasPer library is invoked. This function registers the codecs that are enabled by default.

#### Returns

If successful, zero is returned; otherwise, a nonzero value is returned.

## 9.2 I/O Streams

I/O streams.

#### Macros

- #define [jas\\_stream\\_eof](#)(stream) (((stream)->flags\_ & JAS\_STREAM\_EOF) != 0)  
*Get the EOF indicator for a stream.*
- #define [jas\\_stream\\_error](#)(stream) (((stream)->flags\_ & JAS\_STREAM\_ERR) != 0)  
*Get the error indicator for a stream.*
- #define [jas\\_stream\\_clearerr](#)(stream) ((stream)->flags\_ &= ~(JAS\_STREAM\_ERR | JAS\_STREAM\_EOF))  
*Clear the error indicator for a stream.*
- #define [jas\\_stream\\_getrwlimit](#)(stream) (((const jas\_stream\_t\*)(stream))->rwlimit\_)  
*Get the read/write limit for a stream.*
- #define [jas\\_stream\\_getrwcoun](#)t(stream) (((const jas\_stream\_t\*)(stream))->rwcoun\_t\_)  
*Get the read/write count for a stream.*
- #define [jas\\_stream\\_getc](#)(stream) jas\_stream\_getc\_func(stream)  
*jas\_stream\_getc Read a character from a stream.*
- #define [jas\\_stream\\_putc](#)(stream, c) jas\_stream\_putc\_func(stream, c)  
*jas\_stream\_putc Write a character to a stream.*
- #define [jas\\_stream\\_peekc](#)(stream)  
*Look at the next character to be read from a stream without actually removing the character from the stream.*



## Functions

- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_fopen](#) (`const char *filename`, `const char *mode`)  
*Open a file as a stream.*
- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_memopen](#) (`char *buffer`, `int buffer_size`)  
*Open a memory buffer as a stream.*
- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_memopen2](#) (`char *buffer`, `size_t buffer_size`)
- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_fdopen](#) (`int fd`, `const char *mode`)  
*Open a file descriptor as a stream.*
- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_freopen](#) (`const char *path`, `const char *mode`, `FILE *fp`)  
*Open a stdio (i.e., C standard library) stream as a stream.*
- JAS\_DLLEXPORT `jas_stream_t *` [jas\\_stream\\_tmpfile](#) (`void`)  
*Open a temporary file as a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_close](#) (`jas_stream_t *stream`)  
*Close a stream.*
- JAS\_DLLEXPORT `long` [jas\\_stream\\_setrwlimit](#) (`jas_stream_t *stream`, `long rwlimit`)  
*Set the read/write limit for a stream.*
- JAS\_DLLEXPORT `long` [jas\\_stream\\_setrwcoun](#) (`jas_stream_t *stream`, `long rw_count`)  
*Set the read/write count for a stream.*
- JAS\_DLLEXPORT `unsigned` [jas\\_stream\\_read](#) (`jas_stream_t *stream`, `void *buffer`, `unsigned count`)  
*Read characters from a stream into a buffer.*
- JAS\_DLLEXPORT `unsigned` [jas\\_stream\\_peek](#) (`jas_stream_t *stream`, `void *buffer`, `size_t count`)  
*Attempt to retrieve one or more pending characters of input from a stream into a buffer without actually removing the characters from the stream.*
- JAS\_DLLEXPORT `unsigned` [jas\\_stream\\_write](#) (`jas_stream_t *stream`, `const void *buffer`, `unsigned count`)  
*Write characters from a buffer to a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_printf](#) (`jas_stream_t *stream`, `const char *format`,...)  
*Write formatted output to a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_puts](#) (`jas_stream_t *stream`, `const char *s`)  
*Write a string to a stream.*
- JAS\_DLLEXPORT `char *` [jas\\_stream\\_gets](#) (`jas_stream_t *stream`, `char *buffer`, `int buffer_size`)  
*Read a line of input from a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_ungetc](#) (`jas_stream_t *stream`, `int c`)  
*Put a character back on a stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT `int` [jas\\_stream\\_isseekable](#) (`jas_stream_t *stream`)  
*Determine if stream supports seeking.*
- JAS\_DLLEXPORT `long` [jas\\_stream\\_seek](#) (`jas_stream_t *stream`, `long offset`, `int origin`)  
*Set the current position within the stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT `long` [jas\\_stream\\_tell](#) (`jas_stream_t *stream`)  
*Get the current position within the stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_rewind](#) (`jas_stream_t *stream`)  
*Seek to the beginning of a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_flush](#) (`jas_stream_t *stream`)  
*Flush any pending output to a stream.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_copy](#) (`jas_stream_t *destination`, `jas_stream_t *source`, `int count`)  
*Copy data from one stream to another.*
- JAS\_DLLEXPORT `int` [jas\\_stream\\_display](#) (`jas_stream_t *stream`, `FILE *fp`, `int count`)

*Print a hex dump of data read from a stream.*

- JAS\_DLLEXPORT int [jas\\_stream\\_gobble](#) (jas\_stream\_t \*stream, int count)

*Consume (i.e., discard) characters from stream.*

- JAS\_DLLEXPORT int [jas\\_stream\\_pad](#) (jas\_stream\_t \*stream, int count, int value)

*Write a fill character multiple times to a stream.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT long [jas\\_stream\\_length](#) (jas\_stream\_t \*stream)

*Get the size of the file associated with the specified stream.*

## 9.2.1 Detailed Description

I/O streams.

For more detail on I/O streams, please refer to [here](#).

## 9.2.2 Macro Definition Documentation

### 9.2.2.1 [jas\\_stream\\_clearerr](#)

```
#define jas_stream_clearerr(  
    stream )    ((stream)->flags_ &= ~(JAS_STREAM_ERR | JAS_STREAM_EOF))
```

Clear the error indicator for a stream.

Parameters

<i>stream</i>	The stream whose error indicator is to be cleared.
---------------	--

**Todo** TODO/FIXME: Should this macro evaluate to void?

### 9.2.2.2 [jas\\_stream\\_eof](#)

```
#define jas_stream_eof(  
    stream )    (((stream)->flags_ & JAS_STREAM_EOF) != 0)
```

Get the EOF indicator for a stream.

Parameters

<i>stream</i>	The stream whose EOF indicator is to be queried.
---------------	--

**Returns**

The value of the EOF indicator is returned. A nonzero value indicates that the stream has encountered EOF.

**9.2.2.3 jas\_stream\_error**

```
#define jas_stream_error(  
    stream )  (((stream)->flags_ & JAS_STREAM_ERR) != 0)
```

Get the error indicator for a stream.

**Parameters**

<i>stream</i>	The stream whose error indicator is to be queried.
---------------	--

**Returns**

The value of the error indicator is returned. A nonzero value indicates that the stream has encountered an error of some type (such as an I/O error). Note that EOF is not an error.

**9.2.2.4 jas\_stream\_getc**

```
#define jas_stream_getc(  
    stream )  jas_stream_getc_func(stream)
```

`jas_stream_getc` Read a character from a stream.

**9.2.2.5 jas\_stream\_getrwcoun**

```
#define jas_stream_getrwcoun(  
    stream )  (((const jas_stream_t *) (stream))->rwcnt_)
```

Get the read/write count for a stream.

**Parameters**

<i>stream</i>	A pointer to the stream whose read/write count is to be queried.
---------------	--

**Returns**

The read/write count is returned. This operation cannot fail.

### 9.2.2.6 `jas_stream_getrwlimit`

```
#define jas_stream_getrwlimit(  
    stream )  (((const jas_stream_t *) (stream))->rwlimit_)
```

Get the read/write limit for a stream.

#### Parameters

<i>stream</i>	A pointer to the stream whose read/write limit is to be queried.
---------------	--

#### Returns

The read/write limit for the stream is returned. This operation cannot fail. A negative read/write limit indicates no limit (i.e., an limit that is effectively infinite).

### 9.2.2.7 `jas_stream_peekc`

```
#define jas_stream_peekc(  
    stream )
```

#### Value:

```
((stream)->cnt_ <= 0) ? jas_stream_fillbuf(stream, 0) : \  
((int) (* (stream)->ptr_))
```

Look at the next character to be read from a stream without actually removing the character from the stream.

#### Parameters

<i>stream</i>	A pointer to the stream to be examined.
---------------	---

This function examines the next character that would be read from the stream and returns this character without actually removing it from the stream.

#### Returns

If the peek operation fails (e.g., due to EOF or I/O error), EOF is returned. Otherwise, the character that would be next read from the stream is returned.

### 9.2.2.8 `jas_stream_putc`

```
#define jas_stream_putc(  
    stream,  
    c )  jas_stream_putc_func(stream, c)
```

`jas_stream_putc` Write a character to a stream.

### 9.2.3 Function Documentation

#### 9.2.3.1 `jas_stream_close()`

```
JAS_DLLEXPORT int jas_stream_close (
    jas_stream_t * stream )
```

Close a stream.

##### Parameters

<i>stream</i>	A (nonnull) pointer to the stream to be closed.
---------------	---

The close operation will implicitly flush any pending output to the stream before closing. If such a flush operation fails, this will be reflected in the return value of this function. For many systems, it is likely that the main reason that this function can fail is due to an I/O error when flushing buffered output.

##### Returns

If no errors are encountered when closing the stream, 0 is returned. Otherwise, a nonzero value is returned.

#### 9.2.3.2 `jas_stream_copy()`

```
JAS_DLLEXPORT int jas_stream_copy (
    jas_stream_t * destination,
    jas_stream_t * source,
    int count )
```

Copy data from one stream to another.

##### Parameters

<i>destination</i>	A pointer to the stream that is the destination for the copy.
<i>source</i>	A pointer to the stream that is the source for the copy.
<i>count</i>	The number of characters to copy.

The function copies the specified number of characters from the source stream to the destination stream.

##### Returns

Upon success, 0 is returned; otherwise, -1 is returned.

**Todo** TODO/FIXME: count should probably be a `size_t`; return type `ssize_t`?

### 9.2.3.3 `jas_stream_display()`

```
JAS_DLLEXPORT int jas_stream_display (
    jas_stream_t * stream,
    FILE * fp,
    int count )
```

Print a hex dump of data read from a stream.

#### Parameters

<i>stream</i>	A pointer to the stream from which to read data.
<i>fp</i>	A pointer to a stdio stream (i.e., FILE) to which to print the hex dump.
<i>count</i>	The number of characters to include in the hex dump.

This function prints a hex dump of data read from a stream to a stdio stream. This function is most likely to be useful for debugging.

#### Returns

Upon success, 0 is returned. Otherwise, a negative value is returned.

**Todo** TODO/FIXME: should count be unsigned int or size\_t instead of int?

### 9.2.3.4 `jas_stream_fdopen()`

```
JAS_DLLEXPORT jas_stream_t * jas_stream_fdopen (
    int fd,
    const char * mode )
```

Open a file descriptor as a stream.

#### Parameters

<i>fd</i>	The file descriptor of the file to open as a stream.
<i>mode</i>	A pointer to a string specifying the open mode. The format of this string is similar to that of the fdopen function in the C standard library.

#### Returns

Upon success, a pointer to the opened stream is returned. Otherwise, a null pointer is returned.

### 9.2.3.5 `jas_stream_flush()`

```
JAS_DLLEXPORT int jas_stream_flush (
    jas_stream_t * stream )
```

Flush any pending output to a stream.

#### Parameters

<i>stream</i>	A pointer to the stream for which output should be flushed.
---------------	---

The function flushes any buffered output to the underlying file object.

#### Returns

Upon success, zero is returned. Otherwise, a negative value is returned.

### 9.2.3.6 `jas_stream_fopen()`

```
JAS_DLLEXPORT jas_stream_t * jas_stream_fopen (
    const char * filename,
    const char * mode )
```

Open a file as a stream.

#### Parameters

<i>filename</i>	A pointer to the pathname of the file to be opened.
<i>mode</i>	A pointer to the string specifying the open mode. The open mode is similar to that used by the <code>fopen</code> function in the C standard library.

#### Returns

Upon success, a pointer to the opened stream is returned. Otherwise, a null pointer is returned.

### 9.2.3.7 `jas_stream_freopen()`

```
JAS_DLLEXPORT jas_stream_t * jas_stream_freopen (
    const char * path,
    const char * mode,
    FILE * fp )
```

Open a stdio (i.e., C standard library) stream as a stream.

#### Parameters

<i>path</i>	A pointer to a string containing the path of the filename associated with the stdio stream.
<i>mode</i>	A pointer to a string containing the open mode to be used for the (JasPer) stream. This string is similar to that used by the <code>fdopen</code> function in the C standard library.
<i>fp</i>	A pointer to the stdio stream.

It is unspecified whether the open mode specified by *mode* can be changed from the open mode used for opening the *stdio* stream.

#### Returns

Upon success, a pointer to the opened stream is returned. Otherwise, a null pointer is returned.

#### 9.2.3.8 `jas_stream_gets()`

```
JAS_DLLEXPORT char * jas_stream_gets (
    jas_stream_t * stream,
    char * buffer,
    int buffer_size )
```

Read a line of input from a stream.

#### Parameters

<i>stream</i>	A pointer to the stream from which to read input.
<i>buffer</i>	A pointer to the start of the buffer to hold to input to be read.
<i>buffer_size</i>	The size of the buffer in characters.

The function reads a line of input from a stream into a buffer. If a newline character is read, it is placed in the buffer. Since the buffer may be too small to hold the input, this operation can fail due to attempted buffer overrun.

#### Returns

If the operation fails (e.g., due to an I/O error or attempted buffer overrun), a null pointer is returned. Otherwise, *buffer* is returned.

#### 9.2.3.9 `jas_stream_gobble()`

```
JAS_DLLEXPORT int jas_stream_gobble (
    jas_stream_t * stream,
    int count )
```

Consume (i.e., discard) characters from stream.

#### Parameters

<i>stream</i>	A pointer to the stream from which to discard data.
<i>count</i>	The number of characters to discard.

This function reads and discards the specified number of characters from the given stream.



**Returns**

This function returns the number of characters read and discarded. If an error or EOF is encountered, the number of characters read will be less than count. To distinguish EOF from an I/O error, [jas\\_stream\\_eof\(\)](#) and [jas\\_stream\\_error\(\)](#) can be used.

**Warning**

TODO/FIXME: count be size\_t and return type should be ssize\_t

**9.2.3.10 jas\_stream\_isseekable()**

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_stream_isseekable (
    jas_stream_t * stream )
```

Determine if stream supports seeking.

**Parameters**

<i>stream</i>	A pointer to the stream to query.
---------------	-----------------------------------

The function is a predicate that tests if the underlying file object supports seek operations.

**Returns**

If the underlying file object supports seek operations, a (strictly) positive value is returned. Otherwise, 0 is returned.

**9.2.3.11 jas\_stream\_length()**

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT long jas_stream_length (
    jas_stream_t * stream )
```

Get the size of the file associated with the specified stream.

**Parameters**

<i>stream</i>	
---------------	--

This function queries the size (i.e., length) of the underlying file object associated with the specified stream. The specified stream must be seekable.

**Returns**

Upon success, the size of the stream is returned. If an error occurs, a negative value is returned.

**Warning**

TODO/FIXME: the return type should be `ssize_t`?

**9.2.3.12 `jas_stream_memopen()`**

```
JAS_DLLEXPORT jas_stream_t * jas_stream_memopen (
    char * buffer,
    int buffer_size )
```

Open a memory buffer as a stream.

**Parameters**

<i>buffer</i>	A pointer to the buffer to be used to store stream data.
<i>buffer_size</i>	The size of the buffer.

- If `buffer` is 0 and `buffer_size` > 0: a buffer is dynamically allocated with size `buffer_size` and this buffer is not growable.
- If `buffer` is 0 and `buffer_size` is 0: a buffer is dynamically allocated whose size will automatically grow to accommodate the amount of data written.
- If `buffer` is not 0: `buffer_size` (which, in this case, is not currently allowed to be zero) is the size of the (nongrowable) buffer pointed to by `buffer`.

**Warning**

TODO/FIXME: The type of the `buffer_size` parameter will be changed to `size_t` in the future.

TODO/FIXME: In a later release, this function will be changed to have the same prototype as `jas_stream_memopen2`, at which point `jas_stream_memopen2` will be removed.

**9.2.3.13 `jas_stream_memopen2()`**

```
JAS_DLLEXPORT jas_stream_t * jas_stream_memopen2 (
    char * buffer,
    size_t buffer_size )
```

**Warning**

This function will be renamed `jas_stream_memopen` in a future release. Do not use this function.

**9.2.3.14 `jas_stream_pad()`**

```
JAS_DLLEXPORT int jas_stream_pad (
    jas_stream_t * stream,
    int count,
    int value )
```

Write a fill character multiple times to a stream.

## Parameters

<i>stream</i>	A pointer to the stream to which to write.
<i>count</i>	The number of times to write the fill character to the stream.
<i>value</i>	The fill character.

This function writes the given fill character to a stream a specified number of times. If a count of zero is specified, the function should have no effect.

## Returns

The number of times the fill character was written to the stream is returned. If this value is less than the specified count, an error must have occurred.

**Todo** TODO: should the count be `size_t`; return type maybe `size_t`?

**9.2.3.15 `jas_stream_peek()`**

```
JAS_DLLEXPORT unsigned jas_stream_peek (  
    jas_stream_t * stream,  
    void * buffer,  
    size_t count )
```

Attempt to retrieve one or more pending characters of input from a stream into a buffer without actually removing the characters from the stream.

## Parameters

<i>stream</i>	A pointer to the stream from which to retrieve pending input.
<i>buffer</i>	A pointer to the start of the buffer.
<i>count</i>	A count of how many characters to retrieve.

The extent to which one can peek into the stream is limited. Therefore, this function can fail if count is sufficiently large.

## Returns

Returns the number of bytes copied to the given buffer, or 0 on error or EOF.

## Warning

TODO/FIXME: peeking at EOF should be distinguishable from an I/O error

### 9.2.3.16 `jas_stream_printf()`

```
JAS_DLLEXPORT int jas_stream_printf (
    jas_stream_t * stream,
    const char * format,
    ... )
```

Write formatted output to a stream.

#### Parameters

<i>stream</i>	A pointer to the stream to which to write output.
<i>format</i>	A pointer to a format string similar to the printf function in the C standard library.

The function prints the information associated with the format string to the specified stream.

#### Returns

Upon success, the number of characters output to the stream is returned. If an error is encountered, a negative value is returned.

**Todo** I think that the return type of int is okay here. It is consistent with printf and friends.

### 9.2.3.17 `jas_stream_puts()`

```
JAS_DLLEXPORT int jas_stream_puts (
    jas_stream_t * stream,
    const char * s )
```

Write a string to a stream.

#### Parameters

<i>stream</i>	A pointer to the stream to which the string should be written.
<i>s</i>	A pointer to a null-terminated string for output.

The null character is not output.

#### Returns

Upon success, a nonnegative value is returned. Upon failure, a negative value is returned.

### 9.2.3.18 `jas_stream_read()`

```
JAS_DLLEXPORT unsigned jas_stream_read (
    jas_stream_t * stream,
```

```
void * buffer,
unsigned count )
```

Read characters from a stream into a buffer.

#### Parameters

<i>stream</i>	A pointer to the stream from which to read data.
<i>buffer</i>	A pointer to the start of the buffer.
<i>count</i>	A count of the number of characters to read (nominally).

If *count* is zero, the function has no effect (and therefore cannot fail). Otherwise, the function attempts to read *count* characters from the stream *stream* into the buffer starting at *buffer*. The number of characters read can be less than *count*, due to end-of-file (EOF) or an I/O error.

#### Returns

The number of characters read is returned. In the case that the number of characters read is less than *count*, [jas\\_stream\\_eof\(\)](#) and/or [jas\\_stream\\_error\(\)](#) must be used to distinguish between:

1. a failure due to an I/O error
2. a failure due to the read/write limit being exceeded
3. EOF.

TODO/CHECK: can items 1 and 2 be distinguished currently?

#### Warning

TODO/FIXME/CHECK: *jas\_stream\_error* should be true if RWLIMIT exceeded? or need a *jas\_stream\_rwlimit* predicate?

TODO/FIXME: In the future, the type of the *count* parameter and the return type will be changed to *size\_t*.

### 9.2.3.19 *jas\_stream\_rewind()*

```
JAS_DLLEXPORT int jas_stream_rewind (
    jas_stream_t * stream )
```

Seek to the beginning of a stream.

#### Parameters

<i>stream</i>	A pointer to the stream whose position is to be set.
---------------	--

The stream position is set to the start of the stream. This function is equivalent to returning the value of *jas\_stream\_←*  
*seek(stream, 0, SEEK\_SET)*.

**Returns**

Upon success, the new stream position is returned. Otherwise, a negative value is returned.

**9.2.3.20 jas\_stream\_seek()**

```
JAS_DLLEXPORT long jas_stream_seek (
    jas_stream_t * stream,
    long offset,
    int origin )
```

Set the current position within the stream.

**Parameters**

<i>stream</i>	A pointer to the stream for which to set the current position.
<i>offset</i>	The new position for the stream.
<i>origin</i>	The origin to which this new position is relative.

The origin can be SEEK\_CUR, SEEK\_SET, or SEEK\_END in a similar fashion as the fseek function in the C standard library (and the lseek function in POSIX).

**Returns**

Upon success, the new stream position is returned. Upon failure, a negative value is returned.

**9.2.3.21 jas\_stream\_setrwcoun()**

```
JAS_DLLEXPORT long jas_stream_setrwcoun (
    jas_stream_t * stream,
    long rw_count )
```

Set the read/write count for a stream.

**Parameters**

<i>stream</i>	A pointer to the stream whose read/write count is to be set.
<i>rw_count</i>	The new value for the read/write count.

**Returns**

The old value of the read/write count is returned. This operation cannot fail.

**Todo** TODO/FIXME: Should this macro evaluate to void?

### 9.2.3.22 `jas_stream_setrwlimit()`

```
JAS_DLLEXPORT long jas_stream_setrwlimit (
    jas_stream_t * stream,
    long rwlimit )
```

Set the read/write limit for a stream.

#### Parameters

<i>stream</i>	A pointer to the stream whose read/write limit is to be set.
<i>rwlimit</i>	The new value for the read/write limit.

A negative read/write limit is treated as if it were infinity (i.e., there is no read/write limit).

#### Returns

The old read/write limit is returned.

### 9.2.3.23 `jas_stream_tell()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT long jas_stream_tell (
    jas_stream_t * stream )
```

Get the current position within the stream.

#### Parameters

<i>stream</i>	A pointer to the stream whose current position is to be queried.
---------------	--

The current position of the stream is returned.

#### Returns

Upon success, the current stream position is returned. If an error is encountered, a negative value is returned.

### 9.2.3.24 `jas_stream_tmpfile()`

```
JAS_DLLEXPORT jas_stream_t * jas_stream_tmpfile (
    void )
```

Open a temporary file as a stream.

A temporary file is created and opened as a stream. The temporary file is deleted when closed via [jas\\_stream\\_close\(\)](#). Some operating systems provide a mechanism for ensuring that a file is removed when closed. Such functionality may be used by the implementation when available.

#### Returns

Upon success, a pointer to the opened stream is returned. Otherwise, a null pointer is returned.

### 9.2.3.25 `jas_stream_ungetc()`

```
JAS_DLLEXPORT int jas_stream_ungetc (
    jas_stream_t * stream,
    int c )
```

Put a character back on a stream.

#### Parameters

<i>stream</i>	A pointer to the stream to which the character should be put back.
<i>c</i>	The character to put back.

The character *c* (which was presumably read previously from the stream *stream*) is put back on the stream (as if it had not yet been read). In other words, this function undoes the effect of `jas_stream_getc()`. It is unspecified what happens if the character put back was not the one originally read. The number of characters that can be pushed back onto the stream for subsequent reading is limited. Trying to push back too many characters on a stream will result in an error. The approximate limit is given by the value of `JAS_STREAM_MAXPUTBACK`.

#### Returns

Upon success, zero is returned. If the specified character cannot be pushed back, a negative value is returned.

### 9.2.3.26 `jas_stream_write()`

```
JAS_DLLEXPORT unsigned jas_stream_write (
    jas_stream_t * stream,
    const void * buffer,
    unsigned count )
```

Write characters from a buffer to a stream.

#### Parameters

<i>stream</i>	A pointer to the stream to which to write data.
<i>buffer</i>	A pointer to the start of the buffer.
<i>count</i>	A count of the number of characters to write.

If *count* is zero, the function has no effect (and therefore cannot fail). Otherwise, the function will attempt to write *count* characters from the buffer starting at *buffer* to the stream *stream*. The number of characters written can be less than *count* due to an I/O error or the read/write limit being exceeded.

#### Returns

Upon success, the number of characters successfully written is returned. If an error occurs, the value returned will be less than *count*. The `jas_stream_error()` and `jas_stream_rlimit()` function (TODO/CHECK: the latter of which does not currently exist?) can be used to distinguish between:



1. failure due to an I/O error
2. failure due to the read/write limit being exceeded

#### Warning

TODO/FIXME: The type of the count parameter should be `size_t`. The return type should be `size_t`.

## 9.3 Image

### Classes

- struct `jas_image_cmpt_t`  
*Image component class.*
- struct `jas_image_t`  
*Image class.*
- struct `jas_image_cmptparm_t`  
*Component parameters class.*
- struct `jas_image_fmtops_t`  
*Image format-dependent operations.*
- struct `jas_image_fmtinfo_t`  
*Image format information.*

### Macros

- #define `JAS_IMAGE_MAXFMTS` 32  
*The maximum number of image data formats supported.*
- #define `jas_image_width`(image) ((image)->brx\_ - (image)->tlx\_)  
*Get the width of the image in units of the image reference grid.*
- #define `jas_image_height`(image) ((image)->bry\_ - (image)->tly\_)  
*Get the height of the image in units of the image reference grid.*
- #define `jas_image_tlx`(image) ((image)->tlx\_)  
*Get the x-coordinate of the top-left corner of the image bounding box on the reference grid.*
- #define `jas_image_tly`(image) ((image)->tly\_)  
*Get the y-coordinate of the top-left corner of the image bounding box on the reference grid.*
- #define `jas_image_brx`(image) ((image)->brx\_)  
*Get the x-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).*
- #define `jas_image_bry`(image) ((image)->bry\_)  
*Get the y-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).*
- #define `jas_image_numcmpts`(image) ((image)->numcmpts\_)  
*Get the number of image components.*
- #define `jas_image_clrspc`(image) ((image)->clrspc\_)  
*Get the color model used by the image.*
- #define `jas_image_setclrspc`(image, clrspc) ((image)->clrspc\_ = (clrspc))  
*Set the color model for an image.*
- #define `jas_image_cmptwidth`(image, cmptno) ((image)->cmpts\_[cmptno]->width\_)

- *Get the width of a component.*
- #define [jas\\_image\\_cmptheight](#)(image, cmptno) ((image)->cmpts\_[cmptno]->height\_)
- *Get the height of a component.*
- #define [jas\\_image\\_cmptsgnd](#)(image, cmptno) ((image)->cmpts\_[cmptno]->sgnd\_)
- *Get the signedness of the sample data for a component.*
- #define [jas\\_image\\_cmptprec](#)(image, cmptno) ((image)->cmpts\_[cmptno]->prec\_)
- *Get the precision of the sample data for a component.*
- #define [jas\\_image\\_cmptstep](#)(image, cmptno) ((image)->cmpts\_[cmptno]->hstep\_)
- *Get the horizontal subsampling factor for a component.*
- #define [jas\\_image\\_cmptvstep](#)(image, cmptno) ((image)->cmpts\_[cmptno]->vstep\_)
- *Get the vertical subsampling factor for a component.*
- #define [jas\\_image\\_cmpttlx](#)(image, cmptno) ((image)->cmpts\_[cmptno]->tlx\_)
- *Get the x-coordinate of the top-left corner of a component.*
- #define [jas\\_image\\_cmpttly](#)(image, cmptno) ((image)->cmpts\_[cmptno]->tly\_)
- *Get the y-coordinate of the top-left corner of a component.*
- #define [jas\\_image\\_cmptbrx](#)(image, cmptno)
- *Get the x-coordinate of the bottom-right corner of a component (plus "one").*
- #define [jas\\_image\\_cmptbry](#)(image, cmptno)
- *Get the y-coordinate of the bottom-right corner of a component (plus "one").*
- #define [jas\\_image\\_cmprofn](#)(image) ((image)->cmprofn\_)
- *Get the color management profile of an image.*
- #define [jas\\_image\\_setcmprofn](#)(image, cmprofn) ((image)->cmprofn\_ = cmprofn)
- *Set the color management profile for an image.*

## Typedefs

- typedef int\_fast32\_t [jas\\_image\\_coord\\_t](#)
- *Image coordinate.*
- typedef int\_fast16\_t [jas\\_image\\_colorspc\\_t](#)
- *Color space (e.g., RGB, YCbCr).*
- typedef int\_fast32\_t [jas\\_image\\_cmpttype\\_t](#)
- *Component type (e.g., color, opacity).*
- typedef int\_fast16\_t [jas\\_image\\_smpltype\\_t](#)
- *Component sample data format (e.g., real/integer, signedness, precision).*

## Functions

- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_create](#) (unsigned numcmpts, const [jas\\_image\\_cmptparm\\_t](#) \*cmptparms, [jas\\_clrspc\\_t](#) clrspc)
- *Create an image.*
- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_create0](#) (void)
- *Create an "empty" image.*
- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_copy](#) ([jas\\_image\\_t](#) \*image)
- *Clone an image.*
- JAS\_DLLEXPORT void [jas\\_image\\_destroy](#) ([jas\\_image\\_t](#) \*image)
- *Deallocate any resources associated with an image.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT bool [jas\\_image\\_cmpt\\_domains\\_same](#) (const [jas\\_image\\_t](#) \*image)  
*Test if all components are specified at the same positions in space.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT uint\_fast32\_t [jas\\_image\\_rawsize](#) (const [jas\\_image\\_t](#) \*image)  
*Get the raw size of an image (i.e., the nominal size of the image without any compression).*
- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_decode](#) ([jas\\_stream\\_t](#) \*in, int fmt, const char \*optstr)  
*Create an image from a stream in some specified format.*
- JAS\_DLLEXPORT int [jas\\_image\\_encode](#) ([jas\\_image\\_t](#) \*image, [jas\\_stream\\_t](#) \*out, int fmt, const char \*optstr)  
*Write an image to a stream in a specified format.*
- JAS\_DLLEXPORT int [jas\\_image\\_readcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, [jas\\_matrix\\_t](#) \*data)  
*Read a rectangular region of an image component.*
- JAS\_DLLEXPORT int [jas\\_image\\_writecmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, const [jas\\_matrix\\_t](#) \*data)  
*Write a rectangular region of an image component.*
- JAS\_DLLEXPORT void [jas\\_image\\_delcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno)  
*Delete a component from an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_addcmpt](#) ([jas\\_image\\_t](#) \*image, int cmptno, const [jas\\_image\\_cmptparm\\_t](#) \*cmptparm)  
*Add a component to an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_copycmpt](#) ([jas\\_image\\_t](#) \*dstimage, unsigned dstcmptno, [jas\\_image\\_t](#) \*srcimage, unsigned srccmptno)  
*Copy a component from one image to another.*
- JAS\_DLLEXPORT int [jas\\_image\\_depalettize](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned numlutents, const int\_fast32\_t \*lutents, unsigned dtype, unsigned newcmptno)  
*Depalettize an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_readcmptsample](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned x, unsigned y)  
*Read a component sample for an image.*
- JAS\_DLLEXPORT void [jas\\_image\\_writecmptsample](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned x, unsigned y, int\_fast32\_t v)  
*Write a component sample for an image.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_getcmptbytype](#) (const [jas\\_image\\_t](#) \*image, [jas\\_image\\_cmpttype\\_t](#) ctype)  
*Get an image component by its type.*
- JAS\_DLLEXPORT void [jas\\_image\\_clearfmts](#) (void)  
*Clear the table of image formats.*
- JAS\_DLLEXPORT int [jas\\_image\\_addfmt](#) (int id, const char \*name, const char \*ext, const char \*desc, const [jas\\_image\\_fmtops\\_t](#) \*ops)  
*Add entry to table of image formats.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_strtofmt](#) (const char \*s)  
*Get the ID for the image format with the specified name.*
- JAS\_ATTRIBUTE\_CONST JAS\_DLLEXPORT const char \* [jas\\_image\\_fmtostr](#) (int fmt)  
*Get the name of the image format with the specified ID.*
- JAS\_ATTRIBUTE\_CONST JAS\_DLLEXPORT const [jas\\_image\\_fmtinfo\\_t](#) \* [jas\\_image\\_lookupfmtbyid](#) (int id)  
*Lookup image format information by the format ID.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT const [jas\\_image\\_fmtinfo\\_t](#) \* [jas\\_image\\_lookupfmtbyname](#) (const char \*name)  
*Lookup image format information by the format name.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_fmtfromname](#) (const char \*filename)  
*Guess the format of an image file based on its name.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_getfmt](#) (jas\_stream\_t \*in)  
*Get the format of image data in a stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_ishomosamp](#) (const [jas\\_image\\_t](#) \*image)  
*???*
- JAS\_DLLEXPORT int [jas\\_image\\_sampcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned newcmptno, [jas\\_image\\_coord\\_t](#) ho, [jas\\_image\\_coord\\_t](#) vo, [jas\\_image\\_coord\\_t](#) hs, [jas\\_image\\_coord\\_t](#) vs, int sgnd, unsigned prec)  
*???*
- JAS\_DLLEXPORT int [jas\\_image\\_writecmpt2](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, const long \*buf)  
*Write sample data in a component of an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_readcmpt2](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, long \*buf)  
*Read sample data in a component of an image.*
- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_chclrspc](#) ([jas\\_image\\_t](#) \*image, const [jas\\_cmprof\\_t](#) \*outprof, [jas\\_cmxfm\\_intent\\_t](#) intent)  
*Change the color space for an image.*
- JAS\_DLLEXPORT void [jas\\_image\\_dump](#) ([jas\\_image\\_t](#) \*image, FILE \*out)  
*Dump the information for an image (for debugging).*

### 9.3.1 Detailed Description

### 9.3.2 Macro Definition Documentation

#### 9.3.2.1 [jas\\_image\\_brx](#)

```
#define jas_image_brx(  
    image )    ((image)->brx_)
```

Get the x-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).

#### 9.3.2.2 [jas\\_image\\_bry](#)

```
#define jas_image_bry(  
    image )    ((image)->bry_)
```

Get the y-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).

#### 9.3.2.3 [jas\\_image\\_clrspc](#)

```
#define jas_image_clrspc(  
    image )    ((image)->clrspc_)
```

Get the color model used by the image.

### 9.3.2.4 jas\_image\_cmprof

```
#define jas_image_cmprof(  
    image ) ((image)->cmprof_)
```

Get the color management profile of an image.

### 9.3.2.5 jas\_image\_cmptbrx

```
#define jas_image_cmptbrx(  
    image,  
    cmptno )
```

**Value:**

```
((image)->cmpts_[cmptno]->tlx_ + (image)->cmpts_[cmptno]->width_ * \  
 (image)->cmpts_[cmptno]->hstep_)
```

Get the x-coordinate of the bottom-right corner of a component (plus "one").

### 9.3.2.6 jas\_image\_cmptbry

```
#define jas_image_cmptbry(  
    image,  
    cmptno )
```

**Value:**

```
((image)->cmpts_[cmptno]->tly_ + (image)->cmpts_[cmptno]->height_ * \  
 (image)->cmpts_[cmptno]->vstep_)
```

Get the y-coordinate of the bottom-right corner of a component (plus "one").

### 9.3.2.7 jas\_image\_cmptheight

```
#define jas_image_cmptheight(  
    image,  
    cmptno ) ((image)->cmpts_[cmptno]->height_)
```

Get the height of a component.

### 9.3.2.8 jas\_image\_cmptstep

```
#define jas_image_cmptstep(  
    image,  
    cmptno ) ((image)->cmpts_[cmptno]->hstep_)
```

Get the horizontal subsampling factor for a component.

### 9.3.2.9 jas\_image\_cmptprec

```
#define jas_image_cmptprec(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->prec_)
```

Get the precision of the sample data for a component.

### 9.3.2.10 jas\_image\_cmptsgnd

```
#define jas_image_cmptsgnd(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->sgnd_)
```

Get the signedness of the sample data for a component.

### 9.3.2.11 jas\_image\_cmpttlx

```
#define jas_image_cmpttlx(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->tlx_)
```

Get the x-coordinate of the top-left corner of a component.

### 9.3.2.12 jas\_image\_cmpttly

```
#define jas_image_cmpttly(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->tly_)
```

Get the y-coordinate of the top-left corner of a component.

### 9.3.2.13 jas\_image\_cmptvstep

```
#define jas_image_cmptvstep(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->vstep_)
```

Get the vertical subsampling factor for a component.

### 9.3.2.14 jas\_image\_cmptwidth

```
#define jas_image_cmptwidth(  
    image,  
    cmptno )  ((image)->cmpts_[cmptno]->width_)
```

Get the width of a component.

### 9.3.2.15 `jas_image_height`

```
#define jas_image_height(  
    image )  ((image)->bry_ - (image)->tly_)
```

Get the height of the image in units of the image reference grid.

### 9.3.2.16 `JAS_IMAGE_MAXFMTS`

```
#define JAS_IMAGE_MAXFMTS 32
```

The maximum number of image data formats supported.

### 9.3.2.17 `jas_image_numcmpts`

```
#define jas_image_numcmpts(  
    image )  ((image)->numcmpts_)
```

Get the number of image components.

### 9.3.2.18 `jas_image_setclrspc`

```
#define jas_image_setclrspc(  
    image,  
    clrspc )  ((image)->clrspc_ = (clrspc))
```

Set the color model for an image.

### 9.3.2.19 `jas_image_setcmprof`

```
#define jas_image_setcmprof(  
    image,  
    cmprof )  ((image)->cmprof_ = cmprof)
```

Set the color management profile for an image.

### 9.3.2.20 `jas_image_tlx`

```
#define jas_image_tlx(  
    image )  ((image)->tlx_)
```

Get the x-coordinate of the top-left corner of the image bounding box on the reference grid.

### 9.3.2.21 `jas_image_tly`

```
#define jas_image_tly(  
    image )    ((image)->tly_)
```

Get the y-coordinate of the top-left corner of the image bounding box on the reference grid.

### 9.3.2.22 `jas_image_width`

```
#define jas_image_width(  
    image )    ((image)->brx_ - (image)->tlx_)
```

Get the width of the image in units of the image reference grid.

## 9.3.3 Typedef Documentation

### 9.3.3.1 `jas_image_cmpttype_t`

```
typedef int_fast32_t jas_image_cmpttype_t
```

Component type (e.g., color, opacity).

### 9.3.3.2 `jas_image_colorspc_t`

```
typedef int_fast16_t jas_image_colorspc_t
```

Color space (e.g., RGB, YCbCr).

### 9.3.3.3 `jas_image_coord_t`

```
typedef int_fast32_t jas_image_coord_t
```

Image coordinate.

### 9.3.3.4 `jas_image_smpltype_t`

```
typedef int_fast16_t jas_image_smpltype_t
```

Component sample data format (e.g., real/integer, signedness, precision).



## 9.3.4 Function Documentation

### 9.3.4.1 `jas_image_addcmpt()`

```
JAS_DLLEXPORT int jas_image_addcmpt (
    jas_image_t * image,
    int cmptno,
    const jas_image_cmptparm_t * cmptparm )
```

Add a component to an image.

### 9.3.4.2 `jas_image_addfmt()`

```
JAS_DLLEXPORT int jas_image_addfmt (
    int id,
    const char * name,
    const char * ext,
    const char * desc,
    const jas_image_fmtops_t * ops )
```

Add entry to table of image formats.

### 9.3.4.3 `jas_image_chclrspc()`

```
JAS_DLLEXPORT jas_image_t * jas_image_chclrspc (
    jas_image_t * image,
    const jas_cmprof_t * outprof,
    jas_cmxform_intent_t intent )
```

Change the color space for an image.

### 9.3.4.4 `jas_image_clearfmts()`

```
JAS_DLLEXPORT void jas_image_clearfmts (
    void )
```

Clear the table of image formats.

### 9.3.4.5 `jas_image_cmpt_domains_same()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT bool jas_image_cmpt_domains_same (
    const jas_image_t * image )
```

Test if all components are specified at the same positions in space.

#### 9.3.4.6 jas\_image\_copy()

```
JAS_DLLEXPORT jas_image_t * jas_image_copy (
    jas_image_t * image )
```

Clone an image.

#### 9.3.4.7 jas\_image\_copycmt()

```
JAS_DLLEXPORT int jas_image_copycmt (
    jas_image_t * dstimage,
    unsigned dstcmtptno,
    jas_image_t * srcimage,
    unsigned srccmtptno )
```

Copy a component from one image to another.

#### 9.3.4.8 jas\_image\_create()

```
JAS_DLLEXPORT jas_image_t * jas_image_create (
    unsigned numcmpts,
    const jas_image_cmptparm_t * cmptparms,
    jas_clrspc_t clrspc )
```

Create an image.

#### 9.3.4.9 jas\_image\_create0()

```
JAS_DLLEXPORT jas_image_t * jas_image_create0 (
    void )
```

Create an "empty" image.

#### 9.3.4.10 jas\_image\_decode()

```
JAS_DLLEXPORT jas_image_t * jas_image_decode (
    jas_stream_t * in,
    int fmt,
    const char * optstr )
```

Create an image from a stream in some specified format.

#### 9.3.4.11 `jas_image_delcmt()`

```
JAS_DLLEXPORT void jas_image_delcmt (
    jas_image_t * image,
    unsigned cmtno )
```

Delete a component from an image.

#### 9.3.4.12 `jas_image_depalettize()`

```
JAS_DLLEXPORT int jas_image_depalettize (
    jas_image_t * image,
    unsigned cmtno,
    unsigned numlutents,
    const int_fast32_t * lutents,
    unsigned dtype,
    unsigned newcmtno )
```

Depalettize an image.

#### 9.3.4.13 `jas_image_destroy()`

```
JAS_DLLEXPORT void jas_image_destroy (
    jas_image_t * image )
```

Deallocate any resources associated with an image.

#### 9.3.4.14 `jas_image_dump()`

```
JAS_DLLEXPORT void jas_image_dump (
    jas_image_t * image,
    FILE * out )
```

Dump the information for an image (for debugging).

#### 9.3.4.15 `jas_image_encode()`

```
JAS_DLLEXPORT int jas_image_encode (
    jas_image_t * image,
    jas_stream_t * out,
    int fmt,
    const char * optstr )
```

Write an image to a stream in a specified format.

#### 9.3.4.16 `jas_image_fmtfromname()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_image_fmtfromname (
    const char * filename )
```

Guess the format of an image file based on its name.

#### 9.3.4.17 `jas_image_fmtostr()`

```
JAS_ATTRIBUTE_CONST JAS_DLLEXPORT const char * jas_image_fmtostr (
    int fmt )
```

Get the name of the image format with the specified ID.

#### 9.3.4.18 `jas_image_getcmptbytype()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_image_getcmptbytype (
    const jas_image_t * image,
    jas_image_cmpttype_t ctype )
```

Get an image component by its type.

#### 9.3.4.19 `jas_image_getfmt()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_image_getfmt (
    jas_stream_t * in )
```

Get the format of image data in a stream.

#### 9.3.4.20 `jas_image_ishomosamp()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_image_ishomosamp (
    const jas_image_t * image )
```

???

#### 9.3.4.21 `jas_image_lookupfmtbyid()`

```
JAS_ATTRIBUTE_CONST JAS_DLLEXPORT const jas_image_fmtinfo_t * jas_image_lookupfmtbyid (
    int id )
```

Lookup image format information by the format ID.

#### 9.3.4.22 `jas_image_lookupfmtbyname()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT const jas\_image\_fmtinfo\_t * jas_image_lookupfmtbyname (
    const char * name )
```

Lookup image format information by the format name.

#### 9.3.4.23 `jas_image_rawsize()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT uint_fast32_t jas_image_rawsize (
    const jas\_image\_t * image )
```

Get the raw size of an image (i.e., the nominal size of the image without any compression).

#### 9.3.4.24 `jas_image_readcmpt()`

```
JAS_DLLEXPORT int jas_image_readcmpt (
    jas\_image\_t * image,
    unsigned cmptno,
    jas\_image\_coord\_t x,
    jas\_image\_coord\_t y,
    jas\_image\_coord\_t width,
    jas\_image\_coord\_t height,
    jas\_matrix\_t * data )
```

Read a rectangular region of an image component.

The position and size of the rectangular region to be read is specified relative to the component's coordinate system.

#### 9.3.4.25 `jas_image_readcmpt2()`

```
JAS_DLLEXPORT int jas_image_readcmpt2 (
    jas\_image\_t * image,
    unsigned cmptno,
    jas\_image\_coord\_t x,
    jas\_image\_coord\_t y,
    jas\_image\_coord\_t width,
    jas\_image\_coord\_t height,
    long * buf )
```

Read sample data in a component of an image.

#### 9.3.4.26 `jas_image_readcmptsample()`

```
JAS_DLLEXPORT int jas_image_readcmptsample (
    jas\_image\_t * image,
    unsigned cmptno,
    unsigned x,
    unsigned y )
```

Read a component sample for an image.

#### 9.3.4.27 `jas_image_sampcmpt()`

```
JAS_DLLEXPORT int jas_image_sampcmpt (
    jas_image_t * image,
    unsigned cmptno,
    unsigned newcmptno,
    jas_image_coord_t ho,
    jas_image_coord_t vo,
    jas_image_coord_t hs,
    jas_image_coord_t vs,
    int sgnd,
    unsigned prec )
```

???

#### 9.3.4.28 `jas_image_strtofmt()`

```
JAS_ATTRIBUTE_PURE JAS_DLLEXPORT int jas_image_strtofmt (
    const char * s )
```

Get the ID for the image format with the specified name.

#### 9.3.4.29 `jas_image_writecmpt()`

```
JAS_DLLEXPORT int jas_image_writecmpt (
    jas_image_t * image,
    unsigned cmptno,
    jas_image_coord_t x,
    jas_image_coord_t y,
    jas_image_coord_t width,
    jas_image_coord_t height,
    const jas_matrix_t * data )
```

Write a rectangular region of an image component.

#### 9.3.4.30 `jas_image_writecmpt2()`

```
JAS_DLLEXPORT int jas_image_writecmpt2 (
    jas_image_t * image,
    unsigned cmptno,
    jas_image_coord_t x,
    jas_image_coord_t y,
    jas_image_coord_t width,
    jas_image_coord_t height,
    const long * buf )
```

Write sample data in a component of an image.

#### 9.3.4.31 `jas_image_writemptsample()`

```
JAS_DLLEXPORT void jas_image_writemptsample (
    jas_image_t * image,
    unsigned cmptno,
    unsigned x,
    unsigned y,
    int_fast32_t v )
```

Write a component sample for an image.





# Chapter 10

## Class Documentation

### 10.1 `jas_image_cmpt_t` Struct Reference

Image component class.

```
#include <jas_image.h>
```

#### 10.1.1 Detailed Description

Image component class.

The documentation for this struct was generated from the following file:

- [jas\\_image.h](#)

### 10.2 `jas_image_cmptparm_t` Struct Reference

Component parameters class.

```
#include <jas_image.h>
```

#### 10.2.1 Detailed Description

Component parameters class.

This data type exists solely/mainly for the purposes of the `jas_image_create` function.

The documentation for this struct was generated from the following file:

- [jas\\_image.h](#)

## 10.3 `jas_image_fmtinfo_t` Struct Reference

Image format information.

```
#include <jas_image.h>
```

### 10.3.1 Detailed Description

Image format information.

The documentation for this struct was generated from the following file:

- [jas\\_image.h](#)

## 10.4 `jas_image_fmtops_t` Struct Reference

Image format-dependent operations.

```
#include <jas_image.h>
```

### 10.4.1 Detailed Description

Image format-dependent operations.

The documentation for this struct was generated from the following file:

- [jas\\_image.h](#)

## 10.5 `jas_image_t` Struct Reference

Image class.

```
#include <jas_image.h>
```

### 10.5.1 Detailed Description

Image class.

The documentation for this struct was generated from the following file:

- [jas\\_image.h](#)

# Chapter 11

## File Documentation

### 11.1 bmp\_cod.h

```
00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
```

```

00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Windows Bitmap File Library
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef BMP_COD_H
00071 #define BMP_COD_H
00072
00073 /*****
00074  * Includes.
00075  *****/
00076
00077 #include "jasper/jas_types.h"
00078
00079 /*****
00080  * Constants and macros.
00081  *****/
00082
00083 #define BMP_MAGIC          0x4d42
00084 /* The signature for a BMP file. */
00085
00086 #define BMP_HDRLEN        14
00087 /* The nominal header length. */
00088
00089 #define BMP_INFOLEN        40
00090 /* The nominal info length. */
00091
00092 #define BMP_PALLEN(info)   ((info)->numcolors * 4)
00093 /* The length of the palette. */
00094
00095 #define BMP_HASPAL(info)   ((info)->numcolors > 0)
00096 /* Is this a palettized image? */
00097
00098 /* Encoding types. */
00099 #define BMP_ENC_RGB        0 /* No special encoding. */
00100 #define BMP_ENC_RLE8        1 /* Run length encoding. */
00101 #define BMP_ENC_RLE4        2 /* Run length encoding. */
00102
00103 /*****
00104  * Types.
00105  *****/
00106
00107 /* BMP header. */
00108 typedef struct {
00109
00110     int_fast16_t magic;
00111     /* The signature (a.k.a. the magic number). */
00112
00113     int_fast32_t siz;
00114     /* The size of the file in 32-bit words. */
00115
00116     int_fast16_t reserved1;
00117     /* Ask Bill Gates what this is all about. */
00118
00119     int_fast16_t reserved2;
00120     /* Ditto. */
00121
00122     int_fast32_t off;
00123     /* The offset of the bitmap data from the bitmap file header in bytes. */
00124
00125 } bmp_hdr_t;
00126
00127 /* Palette entry. */
00128 typedef struct {
00129
00130     int_fast16_t red;
00131     /* The red component. */
00132

```

```

00133         int_fast16_t grn;
00134         /* The green component. */
00135
00136         int_fast16_t blu;
00137         /* The blue component. */
00138
00139         int_fast16_t res;
00140         /* Reserved. */
00141
00142     } bmp_palent_t;
00143
00144     /* BMP info. */
00145     typedef struct {
00146
00147         int_fast32_t len;
00148         /* The length of the bitmap information header in bytes. */
00149
00150         int_fast32_t width;
00151         /* The width of the bitmap in pixels. */
00152
00153         int_fast32_t height;
00154         /* The height of the bitmap in pixels. */
00155
00156         int_fast8_t topdown;
00157         /* The bitmap data is specified in top-down order. */
00158
00159         int_fast16_t numplanes;
00160         /* The number of planes. This must be set to a value of one. */
00161
00162         int_fast16_t depth;
00163         /* The number of bits per pixel. */
00164
00165         int_fast32_t enctype;
00166         /* The type of compression used. */
00167
00168         int_fast32_t siz;
00169         /* The size of the image in bytes. */
00170
00171         int_fast32_t hres;
00172         /* The horizontal resolution in pixels/metre. */
00173
00174         int_fast32_t vres;
00175         /* The vertical resolution in pixels/metre. */
00176
00177         int_fast32_t numcolors;
00178         /* The number of color indices used by the bitmap. */
00179
00180         int_fast32_t mincolors;
00181         /* The number of color indices important for displaying the bitmap. */
00182
00183         bmp_palent_t *palents;
00184         /* The colors should be listed in order of importance. */
00185
00186     } bmp_info_t;
00187
00188     /*****
00189     * Functions and macros.
00190     *****/
00191
00192     #define bmp_issupported(hdr, info) \
00193         ((hdr)->magic == BMP_MAGIC && !(hdr)->reserved1 && \
00194          !(hdr)->reserved2 && (info)->numplanes == 1 && \
00195          ((info)->depth == 8 || (info)->depth == 24) && \
00196          (info)->enctype == BMP_ENC_RGB)
00197     /* Is this type of BMP file supported? */
00198
00199     #define bmp_haspal(info) \
00200         ((info)->depth == 8)
00201     /* Is there a palette? */
00202
00203     int bmp_numcmpts(bmp_info_t *info);
00204     /* Get the number of components. */
00205
00206     bmp_info_t *bmp_info_create(void);
00207     /* Create BMP information. */
00208
00209     void bmp_info_destroy(bmp_info_t *info);
00210     /* Destroy BMP information. */
00211
00212     int bmp_isgrayscalepal(bmp_palent_t *palents, int numpalents);
00213     /* Does the specified palette correspond to a grayscale image? */

```

```
00214
00215 #endif
```

## 11.2 bmp\_enc.h

```
00001 /*
00002  * Copyright (c) 2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef BMP_ENC_H
00063 #define BMP_ENC_H
00064
00065 typedef struct {
00066
00067     int numcmpts;
00068     int cmpts[4];
00069
00070 } bmp_enc_t;
00071
00072 #endif
```

## 11.3 jas\_cm.h File Reference

JasPer Color Management.

```
#include <jasper/jas_config.h>
#include <jasper/jas_icc.h>
```

### 11.3.1 Detailed Description

JasPer Color Management.

## 11.4 jas\_cm.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 2002-2003 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
```

```

00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00067 #ifndef JAS_CM_H
00068 #define JAS_CM_H
00069
00070 /* The configuration header file should be included first. */
00071 #include <jasper/jas_config.h>
00072
00073 #include <jasper/jas_icc.h>
00074
00075 #ifdef __cplusplus
00076 extern "C" {
00077 #endif
00078
00079 typedef unsigned jas_clrspc_t;
00080
00081 /* transform operations */
00082 typedef enum {
00083     JAS_CMXFORM_OP_FWD = 0,
00084     JAS_CMXFORM_OP_REV = 1,
00085     JAS_CMXFORM_OP_PROOF = 2,
00086     JAS_CMXFORM_OP_GAMUT = 3,
00087 } jas_cmxform_op_t;
00088
00089 /* rendering intents */
00090 typedef enum {
00091     JAS_CMXFORM_INTENT_PER = 0,
00092     JAS_CMXFORM_INTENT_RELCLR = 1,
00093     JAS_CMXFORM_INTENT_ABSCLR = 2,
00094     JAS_CMXFORM_INTENT_SAT = 3,
00095 } jas_cmxform_intent_t;
00096
00097 #define JAS_CMXFORM_NUMINTENTS 4
00098
00099 typedef enum {
00100     JAS_CMXFORM_OPTM_SPEED = 0,
00101     JAS_CMXFORM_OPTM_SIZE = 1,
00102     JAS_CMXFORM_OPTM_ACC = 2,
00103 } jas_cmxform_optm_t;
00104
00105
00106 #define jas_clrspc_create(fam, mbr) (((fam) < 8) | (mbr))
00107 #define jas_clrspc_fam(clrspc) ((clrspc) >> 8)
00108 #define jas_clrspc_mbr(clrspc) ((clrspc) & 0xff)
00109 #define jas_clrspc_isgeneric(clrspc) (!jas_clrspc_mbr(clrspc))
00110 #define jas_clrspc_isunknown(clrspc) ((clrspc) & JAS_CLRSPC_UNKNOWNMASK)
00111
00112 #define JAS_CLRSPC_UNKNOWNMASK 0x4000
00113
00114 /* color space families */
00115 #define JAS_CLRSPC_FAM_UNKNOWN 0
00116 #define JAS_CLRSPC_FAM_XYZ 1
00117 #define JAS_CLRSPC_FAM_LAB 2
00118 #define JAS_CLRSPC_FAM_GRAY 3
00119 #define JAS_CLRSPC_FAM_RGB 4
00120 #define JAS_CLRSPC_FAM_YCBCR 5
00121
00122 /* specific color spaces */
00123 #define JAS_CLRSPC_UNKNOWN JAS_CLRSPC_UNKNOWNMASK
00124 #define JAS_CLRSPC_CIEXYZ jas_clrspc_create(JAS_CLRSPC_FAM_XYZ, 1)
00125 #define JAS_CLRSPC_CIELAB jas_clrspc_create(JAS_CLRSPC_FAM_LAB, 1)
00126 #define JAS_CLRSPC_SGRAY jas_clrspc_create(JAS_CLRSPC_FAM_GRAY, 1)
00127 #define JAS_CLRSPC_SRGB jas_clrspc_create(JAS_CLRSPC_FAM_RGB, 1)
00128 #define JAS_CLRSPC_SYCBCR jas_clrspc_create(JAS_CLRSPC_FAM_YCBCR, 1)
00129
00130 /* generic color spaces */
00131 #define JAS_CLRSPC_GENRGB jas_clrspc_create(JAS_CLRSPC_FAM_RGB, 0)
00132 #define JAS_CLRSPC_GENGRAY jas_clrspc_create(JAS_CLRSPC_FAM_GRAY, 0)
00133 #define JAS_CLRSPC_GENYCBCR jas_clrspc_create(JAS_CLRSPC_FAM_YCBCR, 0)
00134
00135 #define JAS_CLRSPC_CHANIND_YCBCR_Y 0
00136 #define JAS_CLRSPC_CHANIND_YCBCR_CB 1
00137 #define JAS_CLRSPC_CHANIND_YCBCR_CR 2

```



```

00138
00139 #define JAS_CLRSPC_CHANIND_RGB_R      0
00140 #define JAS_CLRSPC_CHANIND_RGB_G      1
00141 #define JAS_CLRSPC_CHANIND_RGB_B      2
00142
00143 #define JAS_CLRSPC_CHANIND_GRAY_Y      0
00144
00145 typedef double jas_cmreal_t;
00146
00147 struct jas_cmpxform_s;
00148
00149 typedef struct {
00150     long *buf;
00151     unsigned prec;
00152     int sgnd;
00153     unsigned width;
00154     unsigned height;
00155 } jas_cmcmptfmt_t;
00156
00157 typedef struct {
00158     unsigned numcmpts;
00159     jas_cmcmptfmt_t *cmptfmts;
00160 } jas_cmpixmap_t;
00161
00162 typedef struct {
00163     void (*destroy)(struct jas_cmpxform_s *pxform);
00164     int (*apply)(const struct jas_cmpxform_s *pxform, const jas_cmreal_t *in, jas_cmreal_t *out,
00165         unsigned cnt);
00166     void (*dump)(struct jas_cmpxform_s *pxform);
00167 } jas_cmpxformops_t;
00168
00169 typedef struct {
00170     jas_cmreal_t *data;
00171     unsigned size;
00172 } jas_cmshapmatlut_t;
00173
00174 typedef struct {
00175     int mono;
00176     int order;
00177     int useluts;
00178     int usemat;
00179     jas_cmshapmatlut_t luts[3];
00180     jas_cmreal_t mat[3][4];
00181 } jas_cmshapmat_t;
00182
00183 typedef struct {
00184     int order;
00185 } jas_cmshaplut_t;
00186
00187 typedef struct {
00188     unsigned inclrspc;
00189     unsigned outclrspc;
00190 } jas_cmclrspcconv_t;
00191
00192 #define jas_align_t      double
00193
00194 typedef struct jas_cmpxform_s {
00195     unsigned refcnt;
00196     const jas_cmpxformops_t *ops;
00197     unsigned numinchans;
00198     unsigned numoutchans;
00199     union {
00200         jas_align_t dummy;
00201         jas_cmshapmat_t shapmat;
00202         jas_cmshaplut_t shaplut;
00203         jas_cmclrspcconv_t clrspcconv;
00204     } data;
00205 } jas_cmpxform_t;
00206
00207 typedef struct {
00208     unsigned numpxforms;
00209     unsigned maxpxforms;
00210     jas_cmpxform_t **pxforms;
00211 } jas_cmpxformseq_t;
00212
00213 typedef struct {
00214     unsigned numinchans;
00215     unsigned numoutchans;
00216     jas_cmpxformseq_t *pxformseq;
00217 } jas_cmxfm_t;

```

```

00218 #define JAS_CMPROF_TYPE_DEV      1
00219 #define JAS_CMPROF_TYPE_CLRSPEC   2
00220
00221 #define JAS_CMPROF_NUMPXFORMSEQS   13
00222
00223 typedef struct {
00224     jas_clrspc_t clrspc;
00225     unsigned numchans;
00226     unsigned refclrspc;
00227     unsigned numrefchans;
00228     jas_iccprof_t *iccprof;
00229     jas_cmpxformseq_t *pxformseqs[JAS_CMPROF_NUMPXFORMSEQS];
00230 } jas_cmprof_t;
00231
00232 #if 0
00233 typedef int_fast32_t jas_cmattrname_t;
00234 typedef int_fast32_t jas_cmattrval_t;
00235 typedef int_fast32_t jas_cmattrtype_t;
00236 /* Load a profile. */
00237 int jas_cmprof_load(jas_cmprof_t *prof, jas_stream_t *in, unsigned fmt);
00238 /* Save a profile. */
00239 int jas_cmprof_save(jas_cmprof_t *prof, jas_stream_t *out, unsigned fmt);
00240 /* Set an attribute of a profile. */
00241 int jas_cm_prof_setattr(jas_cm_prof_t *prof, jas_cm_attrname_t name, void *val);
00242 /* Get an attribute of a profile. */
00243 void *jas_cm_prof_getattr(jas_cm_prof_t *prof, jas_cm_attrname_t name);
00244 #endif
00245
00246 JAS_DLLEXPORT jas_cmxform_t *jas_cmxform_create(const jas_cmprof_t *inprof, const jas_cmprof_t *outprof,
00247     const jas_cmprof_t *proofprof, jas_cmxform_op_t op, jas_cmxform_intent_t intent, jas_cmxform_optm_t
00248     optimize);
00249 JAS_DLLEXPORT void jas_cmxform_destroy(jas_cmxform_t *xform);
00250
00251 /* Apply a transform to data. */
00252 JAS_DLLEXPORT int jas_cmxform_apply(const jas_cmxform_t *xform, const jas_cmpixmap_t *in,
00253     jas_cmpixmap_t *out);
00254
00255 /* Create a profile. */
00256 JAS_DLLEXPORT jas_cmprof_t *jas_cmprof_createfromiccprof(const jas_iccprof_t *iccprof);
00257 JAS_DLLEXPORT jas_cmprof_t *jas_cmprof_createfromclrspc(jas_clrspc_t clrspc);
00258
00259 /* Destroy a profile. */
00260 JAS_DLLEXPORT void jas_cmprof_destroy(jas_cmprof_t *prof);
00261
00262 unsigned jas_clrspc_numchans(jas_clrspc_t clrspc);
00263 JAS_DLLEXPORT jas_iccprof_t *jas_iccprof_createfromcmprof(const jas_cmprof_t *prof);
00264
00265 #define jas_cmprof_clrspc(prof) ((prof)->clrspc)
00266 JAS_DLLEXPORT jas_cmprof_t *jas_cmprof_copy(const jas_cmprof_t *prof);
00267
00268 #ifdef __cplusplus
00269 }
00270 #endif
00271
00272 #endif

```

## 11.5 jas\_compiler.h File Reference

Compiler-related macros.

```
#include <jasper/jas_config.h>
```

### 11.5.1 Detailed Description

Compiler-related macros.

## 11.6 jas\_compiler.h

[Go to the documentation of this file.](#)

```

00001 /* __START_OF_JASPER_LICENSE__
00002  *
00003  * Jasper License Version 2.0
00004  *
00005  * Copyright (c) 2001-2006 Michael David Adams
00006  * Copyright (c) 1999-2000 Image Power, Inc.
00007  * Copyright (c) 1999-2000 The University of British Columbia
00008  *
00009  * All rights reserved.
00010  *
00011  * Permission is hereby granted, free of charge, to any person (the
00012  * "User") obtaining a copy of this software and associated documentation
00013  * files (the "Software"), to deal in the Software without restriction,
00014  * including without limitation the rights to use, copy, modify, merge,
00015  * publish, distribute, and/or sell copies of the Software, and to permit
00016  * persons to whom the Software is furnished to do so, subject to the
00017  * following conditions:
00018  *
00019  * 1. The above copyright notices and this permission notice (which
00020  * includes the disclaimer below) shall be included in all copies or
00021  * substantial portions of the Software.
00022  *
00023  * 2. The name of a copyright holder shall not be used to endorse or
00024  * promote products derived from the Software without specific prior
00025  * written permission.
00026  *
00027  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00028  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00029  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00030  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00031  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00032  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00033  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00034  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00035  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00036  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00037  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00038  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00039  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00040  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00041  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00042  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00043  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00044  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00045  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00046  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00047  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00048  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00049  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00050  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00051  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00052  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00053  *
00054  * __END_OF_JASPER_LICENSE__
00055  */
00056
00062 #ifndef JAS_COMPILER_H
00063 #define JAS_COMPILER_H
00064
00065 /* The configuration header file should be included first. */
00066 #include <jasper/jas_config.h>
00067
00068 #ifdef _MSC_VER
00069 #ifndef __cplusplus
00070 #undef inline
00071 #define inline __inline
00072 #endif
00073 #endif
00074
00075 #ifdef __GNUC__
00076 #define JAS_DEPRECATED __attribute__((deprecated))
00077 #define JAS_ATTRIBUTE_CONST __attribute__((const))
00078 #define JAS_ATTRIBUTE_PURE __attribute__((pure))
00079 #define JAS_FORCE_INLINE inline __attribute__((always_inline))
00080 #if __GNUC__ > 4 || (__GNUC__ == 4 && __GNUC_MINOR__ >= 5)
00081 #define JAS_UNREACHABLE() __builtin_unreachable()

```

```

00082 #else
00083 #define JAS_UNREACHABLE()
00084 #endif
00085 #define JAS_LIKELY(x) __builtin_expect (!! (x), 1)
00086 #define JAS_UNLIKELY(x) __builtin_expect (!! (x), 0)
00087 #else
00088 #define JAS_DEPRECATED
00089 #define JAS_ATTRIBUTE_CONST
00090 #define JAS_ATTRIBUTE_PURE
00091 #define JAS_FORCE_INLINE inline
00092 #define JAS_UNREACHABLE()
00093 #define JAS_LIKELY(x) (x)
00094 #define JAS_UNLIKELY(x) (x)
00095 #endif
00096
00097 #ifdef __clang__
00098 #define JAS_ATTRIBUTE_DISABLE_USAN \
00099     __attribute__((no_sanitize("undefined")))
00100 #elif defined(__GNUC__) && __GNUC__ >= 6
00101 #define JAS_ATTRIBUTE_DISABLE_USAN \
00102     __attribute__((no_sanitize_undefined))
00103 #else
00104 #define JAS_ATTRIBUTE_DISABLE_USAN
00105 #endif
00106
00107 #ifdef __has_builtin
00108 #define jas_has_builtin(x) __has_builtin(x)
00109 #else
00110 #define jas_has_builtin(x) 0
00111 #endif
00112
00113 #endif

```

## 11.7 jas\_debug.h File Reference

JasPer Debugging-Related Functionality.

```

#include <jasper/jas_config.h>
#include <stdio.h>

```

### 11.7.1 Detailed Description

JasPer Debugging-Related Functionality.

## 11.8 jas\_debug.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.

```

```

00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef JAS_DEBUG_H
00063 #define JAS_DEBUG_H
00064
00065 /*****
00066  * Includes.
00067  *****/
00068 /* The configuration header file should be included first. */
00069 #include <jasper/jas_config.h>
00070
00071 #include <stdio.h>
00072
00073 #ifdef __cplusplus
00074 extern "C" {
00075 #endif
00076
00077 /*****
00078  * Macros and functions.
00079  *****/
00080
00081 /* Output debugging information to standard error provided that the debug
00082  level is set sufficiently high. */
00083 #if !defined(NDEBUG)
00084 #define JAS_DBGLOG(n, x) \
00085     ((jas_getdbglevel() >= (n)) ? (jas_eprintf x) : 0)
00086 #else
00087 #define JAS_DBGLOG(n, x)
00088 #endif
00089
00090 /* Get the library debug level. */
00091 JAS_ATTRIBUTE_CONST
00092 JAS_DLLEXPORT int jas_getdbglevel(void);
00093
00094 /* Set the library debug level. */

```

```

00101 JAS_DLLEXPORT int jas_setdbglevel(int dbglevel);
00102
00103 /* Perform formatted output to standard error. */
00104 JAS_DLLEXPORT int jas_eprintf(const char *fmt, ...);
00105
00106 /* Dump memory to a stream. */
00107 JAS_DLLEXPORT int jas_memdump(FILE *out, const void *data, size_t len);
00108
00109 /* Warn about use of deprecated functionality. */
00110 JAS_DLLEXPORT void jas_deprecated(const char *s);
00111
00112 /* Convert to a string literal */
00113 #define JAS_STRINGIFY(x) #x
00114
00115 /* Convert to a string literal after macro expansion */
00116 #define JAS_STRINGIFYX(x) JAS_STRINGIFY(x)
00117
00118 #ifdef __cplusplus
00119 }
00120 #endif
00121
00122 #endif

```

## 11.9 jas\_dll.h File Reference

Shared Library Macros.

```
#include <jasper/jas_config.h>
```

### 11.9.1 Detailed Description

Shared Library Macros.

## 11.10 jas\_dll.h

[Go to the documentation of this file.](#)

```

00001 /* __START_OF_JASPER_LICENSE__
00002  *
00003  * JasPer License Version 2.0
00004  *
00005  * Copyright (c) 2001-2006 Michael David Adams
00006  * Copyright (c) 1999-2000 Image Power, Inc.
00007  * Copyright (c) 1999-2000 The University of British Columbia
00008  *
00009  * All rights reserved.
00010  *
00011  * Permission is hereby granted, free of charge, to any person (the
00012  * "User") obtaining a copy of this software and associated documentation
00013  * files (the "Software"), to deal in the Software without restriction,
00014  * including without limitation the rights to use, copy, modify, merge,
00015  * publish, distribute, and/or sell copies of the Software, and to permit
00016  * persons to whom the Software is furnished to do so, subject to the
00017  * following conditions:
00018  *
00019  * 1. The above copyright notices and this permission notice (which
00020  * includes the disclaimer below) shall be included in all copies or
00021  * substantial portions of the Software.
00022  *
00023  * 2. The name of a copyright holder shall not be used to endorse or
00024  * promote products derived from the Software without specific prior
00025  * written permission.
00026  *

```

```

00027  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00028  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00029  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00030  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00031  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00032  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00033  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00034  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00035  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00036  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00037  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00038  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00039  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00040  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00041  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00042  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00043  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00044  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00045  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00046  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00047  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00048  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00049  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00050  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00051  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00052  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00053  *
00054  * __END_OF_JASPER_LICENSE__
00055  */
00056
00062 #ifndef JAS_DLL_H
00063 #define JAS_DLL_H
00064
00065 /* The configuration header file should be included first. */
00066 #include <jasper/jas_config.h>
00067
00068 #if defined(JAS_DLL)
00069     #if defined(_WIN32)
00070         #if defined(JAS_BUILDING_DLL)
00071             #define JAS_DLLEXPORT __declspec(dllexport)
00072         #else
00073             #define JAS_DLLEXPORT __declspec(dllimport)
00074         #endif
00075         #define JAS_DLLLOCAL
00076     #elif defined(JAS_HAVE_VISIBILITY)
00077         #if defined(JAS_BUILDING_DLL)
00078             #define JAS_DLLEXPORT __attribute__((visibility("default")))
00079             #define JAS_DLLLOCAL __attribute__((visibility("hidden")))
00080         #else
00081             #define JAS_DLLEXPORT
00082             #define JAS_DLLLOCAL
00083         #endif
00084     #else
00085         #define JAS_DLLEXPORT
00086         #define JAS_DLLLOCAL
00087     #endif
00088 #else
00089     #define JAS_DLLEXPORT
00090     #define JAS_DLLLOCAL
00091 #endif
00092
00093 #endif

```

## 11.11 jas\_fix.h File Reference

JasPer Fixed-Point Number Class.

```

#include <jasper/jas_config.h>
#include <jasper/jas_types.h>

```

### 11.11.1 Detailed Description

JasPer Fixed-Point Number Class.

## 11.12 jas\_fix.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_FIX_H
00070 #define JAS_FIX_H
00071

```



```

00072 /*****
00073  * Includes.
00074  *****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h> /* IWYU pragma: keep */
00078
00079 #include <jasper/jas_types.h>
00080
00081 #ifdef __cplusplus
00082 extern "C" {
00083 #endif
00084
00085 /*****
00086  * Constants.
00087  *****/
00088
00089 /* The representation of the value zero. */
00090 #define JAS_FIX_ZERO(fix_t, fracbits) \
00091     JAS_INTTOFIX(fix_t, fracbits, 0)
00092
00093 /* The representation of the value one. */
00094 #define JAS_FIX_ONE(fix_t, fracbits) \
00095     JAS_INTTOFIX(fix_t, fracbits, 1)
00096
00097 /* The representation of the value one half. */
00098 #define JAS_FIX_HALF(fix_t, fracbits) \
00099     (JAS_CAST(fix_t, 1) << ((fracbits) - 1))
00100
00101 /*****
00102  * Conversion operations.
00103  *****/
00104
00105 /* Convert an int to a fixed-point number. */
00106 #define JAS_INTTOFIX(fix_t, fracbits, x) \
00107     (JAS_CAST(fix_t, x) << (fracbits))
00108
00109 /* Convert a fixed-point number to an int. */
00110 #define JAS_FIXTOINT(fix_t, fracbits, x) \
00111     JAS_CAST(int, (x) >> (fracbits))
00112
00113 /* Convert a fixed-point number to a double. */
00114 #define JAS_FIXTODBL(fix_t, fracbits, x) \
00115     (JAS_CAST(double, x) / JAS_FIX_ONE(fix_t, fracbits))
00116
00117 /* Convert a double to a fixed-point number. */
00118 #define JAS_DBLTOFIX(fix_t, fracbits, x) \
00119     JAS_CAST(fix_t, ((x) * JAS_CAST(double, JAS_FIX_ONE(fix_t, fracbits))))
00120
00121 /*****
00122  * Basic arithmetic operations.
00123  * All other arithmetic operations are synthesized from these basic operations.
00124  * There are three macros for each type of arithmetic operation.
00125  * One macro always performs overflow/underflow checking, one never performs
00126  * overflow/underflow checking, and one is generic with its behavior
00127  * depending on compile-time flags.
00128  * Only the generic macros should be invoked directly by application code.
00129  *****/
00130
00131 /* Calculate the sum of two fixed-point numbers. */
00132 #if !defined(DEBUG_OVERFLOW)
00133 #define JAS_FIX_ADD                                JAS_FIX_ADD_FAST
00134 #else
00135 #define JAS_FIX_ADD                                JAS_FIX_ADD_OFLOW
00136 #endif
00137
00138 /* Calculate the sum of two fixed-point numbers without overflow checking. */
00139 #define JAS_FIX_ADD_FAST(fix_t, fracbits, x, y) ((x) + (y))
00140
00141 /* Calculate the sum of two fixed-point numbers with overflow checking. */
00142 #define JAS_FIX_ADD_OFLOW(fix_t, fracbits, x, y) \
00143     ((x) >= 0) ? \
00144         (((y) >= 0) ? ((x) + (y) >= 0 || JAS_FIX_OFLOW(), (x) + (y)) : \
00145          ((x) + (y))) : \
00146         (((y) >= 0) ? ((x) + (y)) : ((x) + (y) < 0 || JAS_FIX_OFLOW(), \
00147          (x) + (y)))
00148
00149 /* Calculate the product of two fixed-point numbers. */
00150 #if !defined(DEBUG_MUL_OVERFLOW)
00151 #define JAS_FIX_MUL                                JAS_FIX_MUL_FAST
00152 #else

```

```

00153 #define JAS_FIX_MUL                                JAS_FIX_MUL_OFLOW
00154 #endif
00155
00156 /* Calculate the product of two fixed-point numbers without overflow
00157    checking. */
00158 #define JAS_FIX_MUL_FAST(fix_t, fracbits, bigfix_t, x, y) \
00159     JAS_CAST(fix_t, (JAS_CAST(bigfix_t, x) * JAS_CAST(bigfix_t, y)) » \
00160         (fracbits))
00161
00162 /* Calculate the product of two fixed-point numbers with overflow
00163    checking. */
00164 #define JAS_FIX_MUL_OFLOW(fix_t, fracbits, bigfix_t, x, y) \
00165     ((JAS_CAST(bigfix_t, x) * JAS_CAST(bigfix_t, y)) » (fracbits)) == \
00166     JAS_CAST(fix_t, (JAS_CAST(bigfix_t, x) * JAS_CAST(bigfix_t, y)) » \
00167         (fracbits))) ? \
00168     JAS_CAST(fix_t, (JAS_CAST(bigfix_t, x) * JAS_CAST(bigfix_t, y)) » \
00169         (fracbits))) : JAS_FIX_OFLOW()
00170
00171 /* Calculate the product of a fixed-point number and an int. */
00172 #if !defined(DEBUG_OVERFLOW)
00173 #define JAS_FIX_MULBYINT                                JAS_FIX_MULBYINT_FAST
00174 #else
00175 #define JAS_FIX_MULBYINT                                JAS_FIX_MULBYINT_OFLOW
00176 #endif
00177
00178 /* Calculate the product of a fixed-point number and an int without overflow
00179    checking. */
00180 #define JAS_FIX_MULBYINT_FAST(fix_t, fracbits, x, y) \
00181     JAS_CAST(fix_t, ((x) * (y)))
00182
00183 /* Calculate the product of a fixed-point number and an int with overflow
00184    checking. */
00185 #define JAS_FIX_MULBYINT_OFLOW(fix_t, fracbits, x, y) \
00186     JAS_FIX_MULBYINT_FAST(fix_t, fracbits, x, y)
00187
00188 /* Calculate the quotient of two fixed-point numbers. */
00189 #if !defined(DEBUG_OVERFLOW)
00190 #define JAS_FIX_DIV                                JAS_FIX_DIV_FAST
00191 #else
00192 #define JAS_FIX_DIV                                JAS_FIX_DIV_UFLOW
00193 #endif
00194
00195 /* Calculate the quotient of two fixed-point numbers without underflow
00196    checking. */
00197 #define JAS_FIX_DIV_FAST(fix_t, fracbits, bigfix_t, x, y) \
00198     JAS_CAST(fix_t, (JAS_CAST(bigfix_t, x) « (fracbits)) / (y))
00199
00200 /* Calculate the quotient of two fixed-point numbers with underflow
00201    checking. */
00202 #define JAS_FIX_DIV_UFLOW(fix_t, fracbits, bigfix_t, x, y) \
00203     JAS_FIX_DIV_FAST(fix_t, fracbits, bigfix_t, x, y)
00204
00205 /* Negate a fixed-point number. */
00206 #if !defined(DEBUG_OVERFLOW)
00207 #define JAS_FIX_NEG                                JAS_FIX_NEG_FAST
00208 #else
00209 #define JAS_FIX_NEG                                JAS_FIX_NEG_OFLOW
00210 #endif
00211
00212 /* Negate a fixed-point number without overflow checking. */
00213 #define JAS_FIX_NEG_FAST(fix_t, fracbits, x) \
00214     (-x)
00215
00216 /* Negate a fixed-point number with overflow checking. */
00217 /* Yes, overflow is actually possible for two's complement representations,
00218    although highly unlikely to occur. */
00219 #define JAS_FIX_NEG_OFLOW(fix_t, fracbits, x) \
00220     ((x) < 0) ? (-x) > 0 || JAS_FIX_OFLOW(), -(x) : -(x))
00221
00222 /* Perform an arithmetic shift left of a fixed-point number. */
00223 #if !defined(DEBUG_OVERFLOW)
00224 #define JAS_FIX_AS_L                                JAS_FIX_AS_L_FAST
00225 #else
00226 #define JAS_FIX_AS_L                                JAS_FIX_AS_L_OFLOW
00227 #endif
00228
00229 /* Perform an arithmetic shift left of a fixed-point number without overflow
00230    checking. */
00231 #define JAS_FIX_AS_L_FAST(fix_t, fracbits, x, n) \
00232     ((x) « (n))
00233

```

```

00234 /* Perform an arithmetic shift left of a fixed-point number with overflow
00235    checking. */
00236 #define JAS_FIX_AS_L_OFLOW(fix_t, fracbits, x, n) \
00237     (((x) << (n)) >> (n)) == (x) || JAS_FIX_OFLOW(), (x) << (n))
00238
00239 /* Perform an arithmetic shift right of a fixed-point number. */
00240 #if !defined(DEBUG_OVERFLOW)
00241 #define JAS_FIX_AS_R          JAS_FIX_AS_R_FAST
00242 #else
00243 #define JAS_FIX_AS_R          JAS_FIX_AS_R_UFLOW
00244 #endif
00245
00246 /* Perform an arithmetic shift right of a fixed-point number without underflow
00247    checking. */
00248 #define JAS_FIX_AS_R_FAST(fix_t, fracbits, x, n) \
00249     ((x) >> (n))
00250
00251 /* Perform an arithmetic shift right of a fixed-point number with underflow
00252    checking. */
00253 #define JAS_FIX_AS_R_UFLOW(fix_t, fracbits, x, n) \
00254     JAS_FIX_AS_R_FAST(fix_t, fracbits, x, n)
00255
00256 /*****
00257  * Other basic arithmetic operations.
00258  *****/
00259
00260 /* Calculate the difference between two fixed-point numbers. */
00261 #define JAS_FIX_SUB(fix_t, fracbits, x, y) \
00262     JAS_FIX_ADD(fix_t, fracbits, x, JAS_FIX_NEG(fix_t, fracbits, y))
00263
00264 /* Add one fixed-point number to another. */
00265 #define JAS_FIX_PLUSEQ(fix_t, fracbits, x, y) \
00266     ((x) = JAS_FIX_ADD(fix_t, fracbits, x, y))
00267
00268 /* Subtract one fixed-point number from another. */
00269 #define JAS_FIX_MINUSEQ(fix_t, fracbits, x, y) \
00270     ((x) = JAS_FIX_SUB(fix_t, fracbits, x, y))
00271
00272 /* Multiply one fixed-point number by another. */
00273 #define JAS_FIX_MULEQ(fix_t, fracbits, bigfix_t, x, y) \
00274     ((x) = JAS_FIX_MUL(fix_t, fracbits, bigfix_t, x, y))
00275
00276 /*****
00277  * Miscellaneous operations.
00278  *****/
00279
00280 /* Calculate the absolute value of a fixed-point number. */
00281 #define JAS_FIX_ABS(fix_t, fracbits, x) \
00282     (((x) >= 0) ? (x) : (JAS_FIX_NEG(fix_t, fracbits, x)))
00283
00284 /* Is a fixed-point number an integer? */
00285 #define JAS_FIX_ISINT(fix_t, fracbits, x) \
00286     (JAS_FIX_FLOOR(fix_t, fracbits, x) == (x))
00287
00288 /* Get the sign of a fixed-point number. */
00289 #define JAS_FIX_SGN(fix_t, fracbits, x) \
00290     ((x) >= 0 ? 1 : (-1))
00291
00292 /*****
00293  * Relational operations.
00294  *****/
00295
00296 /* Compare two fixed-point numbers. */
00297 #define JAS_FIX_CMP(fix_t, fracbits, x, y) \
00298     ((x) > (y) ? 1 : (((x) == (y)) ? 0 : (-1)))
00299
00300 /* Less than. */
00301 #define JAS_FIX_LT(fix_t, fracbits, x, y) \
00302     ((x) < (y))
00303
00304 /* Less than or equal. */
00305 #define JAS_FIX_LTE(fix_t, fracbits, x, y) \
00306     ((x) <= (y))
00307
00308 /* Greater than. */
00309 #define JAS_FIX_GT(fix_t, fracbits, x, y) \
00310     ((x) > (y))
00311
00312 /* Greater than or equal. */
00313 #define JAS_FIX_GTE(fix_t, fracbits, x, y) \
00314     ((x) >= (y))

```

```

00315
00316 /*****
00317  * Rounding functions.
00318  *****/
00319
00320 /* Round a fixed-point number to the nearest integer. */
00321 #define JAS_FIX_ROUND(fix_t, fracbits, x) \
00322     (((x) < 0) ? JAS_FIX_FLOOR(fix_t, fracbits, JAS_FIX_ADD(fix_t, fracbits, \
00323     (x), JAS_FIX_HALF(fix_t, fracbits))) : \
00324     JAS_FIX_NEG(fix_t, fracbits, JAS_FIX_FLOOR(fix_t, fracbits, \
00325     JAS_FIX_ADD(fix_t, fracbits, -(x), JAS_FIX_HALF(fix_t, fracbits))))))
00326
00327 /* Round a fixed-point number to the nearest integer in the direction of
00328  negative infinity (i.e., the floor function). */
00329 #define JAS_FIX_FLOOR(fix_t, fracbits, x) \
00330     ((x) & ~(JAS_FIX_ONE(fix_t, fracbits) - 1))
00331
00332 /*****
00333  * The below macros are for internal library use only. Do not invoke them
00334  * directly in application code.
00335  *****/
00336
00337 /* Handle overflow. */
00338 #define JAS_FIX_OVERFLOW() \
00339     jas_eprintf("overflow error: file %s, line %d\n", __FILE__, __LINE__)
00340
00341 /* Handle underflow. */
00342 #define JAS_FIX_UFLOW() \
00343     jas_eprintf("underflow error: file %s, line %d\n", __FILE__, __LINE__)
00344
00345 #ifdef __cplusplus
00346 }
00347 #endif
00348
00349 #endif

```

## 11.13 jas\_getopt.h File Reference

Command Line Option Parsing Code.

```
#include <jasper/jas_config.h>
```

### 11.13.1 Detailed Description

Command Line Option Parsing Code.

## 11.14 jas\_getopt.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.

```

```

00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_GETOPT_H
00070 #define JAS_GETOPT_H
00071
00072 #ifdef __cplusplus
00073 extern "C" {
00074 #endif
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h>
00078
00079 /*****
00080  * Constants.
00081  *****/
00082 #define JAS_GETOPT_EOF (-1)
00083 #define JAS_GETOPT_ERR '?'
00084
00085 /* option flags. */
00086 #define JAS_OPT_HASARG 0x01 /* option has argument */
00087
00088 /*****
00089  * Types.
00090  *****/
00091 /* Command line option type. */
00092 typedef struct {
00093     int id;
00094     /* The unique identifier for this option. */
00095     const char *name;

```

```

00100      /* The name of this option. */
00101
00102      int flags;
00103      /* option flags. */
00104
00105 } jas_opt_t;
00106
00107 /*****
00108  * External data.
00109  *****/
00110
00111 /* The current option index. */
00112 JAS_DLLEXPORT extern int jas_optind;
00113
00114 /* The current option argument. */
00115 JAS_DLLEXPORT extern const char *jas_optarg;
00116
00117 /* The debug level. */
00118 JAS_DLLEXPORT extern int jas_opterr;
00119
00120 /*****
00121  * Prototypes.
00122  *****/
00123
00124 /* Get the next option. */
00125 JAS_DLLEXPORT int jas_getopt(int argc, char **argv, const jas_opt_t *opts);
00126
00127 #ifdef __cplusplus
00128 }
00129 #endif
00130
00131 #endif

```

## 11.15 jas\_icc.h File Reference

ICC Profile.

```

#include <jasper/jas_config.h>
#include <jasper/jas_types.h>
#include <jasper/jas_stream.h>
#include <stdio.h>

```

### 11.15.1 Detailed Description

ICC Profile.

## 11.16 jas\_icc.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 2002-2003 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.

```

```
00012 * Copyright (c) 1999-2000 The University of British Columbia
00013 *
00014 * All rights reserved.
00015 *
00016 * Permission is hereby granted, free of charge, to any person (the
00017 * "User") obtaining a copy of this software and associated documentation
00018 * files (the "Software"), to deal in the Software without restriction,
00019 * including without limitation the rights to use, copy, modify, merge,
00020 * publish, distribute, and/or sell copies of the Software, and to permit
00021 * persons to whom the Software is furnished to do so, subject to the
00022 * following conditions:
00023 *
00024 * 1. The above copyright notices and this permission notice (which
00025 * includes the disclaimer below) shall be included in all copies or
00026 * substantial portions of the Software.
00027 *
00028 * 2. The name of a copyright holder shall not be used to endorse or
00029 * promote products derived from the Software without specific prior
00030 * written permission.
00031 *
00032 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 #ifndef JAS_ICC_H
00063 #define JAS_ICC_H
00064
00065 /* The configuration header file should be included first. */
00066 #include <jasper/jas_config.h>
00067
00068 #include <jasper/jas_types.h>
00069 #include <jasper/jas_stream.h>
00070
00071 #include <stdio.h>
00072
00073 #ifdef __cplusplus
00074 extern "C" {
00075 #endif
00076
00077 /* Profile file signature. */
00078 #define JAS_ICC_MAGIC 0x61637370
00079
00080 #define JAS_ICC_HDRLEN 128
00081
00082 /* Profile/device class signatures. */
00083 #define JAS_ICC_CLAS_IN 0x73636e72 /* input device */
00084 #define JAS_ICC_CLAS_DPY 0x646e7472 /* display device */
00085 #define JAS_ICC_CLAS_OUT 0x70727472 /* output device */
00086 #define JAS_ICC_CLAS_LNK 0x6c696e6b /* device link */
00087 #define JAS_ICC_CLAS_CNV 0x73706163 /* color space conversion */
00088 #define JAS_ICC_CLAS_ABS 0x61627374 /* abstract */
00089 #define JAS_ICC_CLAS_NAM 0x6e64636c /* named color */
00090
00091 /* Color space signatures. */
00092 #define JAS_ICC_COLORSPC_XYZ 0x58595a20 /* XYZ */
```

```

00098 #define JAS_ICC_COLORSPC_LAB      0x4c616220 /* LAB */
00099 #define JAS_ICC_COLORSPC_LUV      0x4c757620 /* LUV */
00100 #define JAS_ICC_COLORSPC_YCBCR    0x59436272 /* YCbCr */
00101 #define JAS_ICC_COLORSPC_YXY      0x59787920 /* Yxy */
00102 #define JAS_ICC_COLORSPC_RGB      0x52474220 /* RGB */
00103 #define JAS_ICC_COLORSPC_GRAY     0x47524159 /* Gray */
00104 #define JAS_ICC_COLORSPC_HSV      0x48535620 /* HSV */
00105 #define JAS_ICC_COLORSPC_HLS      0x484c5320 /* HLS */
00106 #define JAS_ICC_COLORSPC_CMYK     0x434d594b /* CMYK */
00107 #define JAS_ICC_COLORSPC_CMY      0x434d5920 /* CMY */
00108 #define JAS_ICC_COLORSPC_2        0x32434c52 /* 2 channel color */
00109 #define JAS_ICC_COLORSPC_3        0x33434c52 /* 3 channel color */
00110 #define JAS_ICC_COLORSPC_4        0x34434c52 /* 4 channel color */
00111 #define JAS_ICC_COLORSPC_5        0x35434c52 /* 5 channel color */
00112 #define JAS_ICC_COLORSPC_6        0x36434c52 /* 6 channel color */
00113 #define JAS_ICC_COLORSPC_7        0x37434c52 /* 7 channel color */
00114 #define JAS_ICC_COLORSPC_8        0x38434c52 /* 8 channel color */
00115 #define JAS_ICC_COLORSPC_9        0x39434c52 /* 9 channel color */
00116 #define JAS_ICC_COLORSPC_10       0x41434c52 /* 10 channel color */
00117 #define JAS_ICC_COLORSPC_11       0x42434c52 /* 11 channel color */
00118 #define JAS_ICC_COLORSPC_12       0x43434c52 /* 12 channel color */
00119 #define JAS_ICC_COLORSPC_13       0x44434c52 /* 13 channel color */
00120 #define JAS_ICC_COLORSPC_14       0x45434c52 /* 14 channel color */
00121 #define JAS_ICC_COLORSPC_15       0x46434c52 /* 15 channel color */
00122
00123 /* Profile connection color space (PCS) signatures. */
00124 #define JAS_ICC_REFCOLORSPC_XYZ    0x58595a20 /* CIE XYZ */
00125 #define JAS_ICC_REFCOLORSPC_LAB    0x4c616220 /* CIE Lab */
00126
00127 /* Primary platform signatures. */
00128 #define JAS_ICC_PLATFORM_APPL      0x4150504c /* Apple Computer */
00129 #define JAS_ICC_PLATFORM_MSFT      0x4d534654 /* Microsoft */
00130 #define JAS_ICC_PLATFORM_SGI       0x53474920 /* Silicon Graphics */
00131 #define JAS_ICC_PLATFORM_SUNW      0x53554e57 /* Sun Microsystems */
00132 #define JAS_ICC_PLATFORM_TGNT      0x54474e54 /* Taligent */
00133
00134 /* Profile flags. */
00135 #define JAS_ICC_FLAGS_EMBED        0x01 /* embedded */
00136 #define JAS_ICC_FLAGS_NOSEP        0x02 /* no separate use */
00137
00138 /* Attributes. */
00139 #define JAS_ICC_ATTR_TRANS          0x01 /* transparent */
00140 #define JAS_ICC_ATTR_MATTE         0x02 /* matte */
00141
00142 /* Rendering intents. */
00143 #define JAS_ICC_INTENT_PER          0 /* perceptual */
00144 #define JAS_ICC_INTENT_REL          1 /* relative colorimetric */
00145 #define JAS_ICC_INTENT_SAT          2 /* saturation */
00146 #define JAS_ICC_INTENT_ABS          3 /* absolute colorimetric */
00147
00148 /* Tag signatures. */
00149 #define JAS_ICC_TAG_ATOB0          0x41324230 /* */
00150 #define JAS_ICC_TAG_ATOB1          0x41324231 /* */
00151 #define JAS_ICC_TAG_ATOB2          0x41324232 /* */
00152 #define JAS_ICC_TAG_BLUMATCOL      0x6258595a /* */
00153 #define JAS_ICC_TAG_BLUTRC         0x62545243 /* */
00154 #define JAS_ICC_TAG_BTOA0          0x42324130 /* */
00155 #define JAS_ICC_TAG_BTOA1          0x42324131 /* */
00156 #define JAS_ICC_TAG_BTOA2          0x42324132 /* */
00157 #define JAS_ICC_TAG_CALTIME        0x63616c74 /* */
00158 #define JAS_ICC_TAG_CHARTARGET     0x74617267 /* */
00159 #define JAS_ICC_TAG_CPYRT          0x63707274 /* */
00160 #define JAS_ICC_TAG_CRDINFO        0x63726469 /* */
00161 #define JAS_ICC_TAG_DEVMAKERDESC   0x646d6e64 /* */
00162 #define JAS_ICC_TAG_DEVMODELDESC   0x646d6464 /* */
00163 #define JAS_ICC_TAG_DEVSET         0x64657673 /* */
00164 #define JAS_ICC_TAG_GAMUT          0x67616d74 /* */
00165 #define JAS_ICC_TAG_GRYTRC         0x6b545243 /* */
00166 #define JAS_ICC_TAG_GRNMATCOL      0x6758595a /* */
00167 #define JAS_ICC_TAG_GRNTRC         0x67545243 /* */
00168 #define JAS_ICC_TAG_LUM            0x6c756d69 /* */
00169 #define JAS_ICC_TAG_MEASURE        0x6d656173 /* */
00170 #define JAS_ICC_TAG_MEDIABLKPT     0x626b7074 /* */
00171 #define JAS_ICC_TAG_MEDIAWHIPT     0x77747074 /* */
00172 #define JAS_ICC_TAG_NAMCOLR        0x6e636f6c /* */
00173 #define JAS_ICC_TAG_NAMCOLR2       0x6e636c32 /* */
00174 #define JAS_ICC_TAG_OUTRESP        0x72657370 /* */
00175 #define JAS_ICC_TAG_PREVIEW0       0x70726530 /* */
00176 #define JAS_ICC_TAG_PREVIEW1       0x70726531 /* */
00177 #define JAS_ICC_TAG_PREVIEW2       0x70726532 /* */
00178 #define JAS_ICC_TAG_PROFDESC       0x64657363 /* */

```



```

00179 #define JAS_ICC_TAG_PROFSEQDESC 0x70736571 /* */
00180 #define JAS_ICC_TAG_PSDCRD0 0x70736430 /* */
00181 #define JAS_ICC_TAG_PSCRDD1 0x70736431 /* */
00182 #define JAS_ICC_TAG_PSCRDD2 0x70736432 /* */
00183 #define JAS_ICC_TAG_PSCRDD3 0x70736433 /* */
00184 #define JAS_ICC_TAG_PS2CSA 0x70733273 /* */
00185 #define JAS_ICC_TAG_PS2RENINTENT 0x70733269 /* */
00186 #define JAS_ICC_TAG_REDMATCOL 0x7258595a /* */
00187 #define JAS_ICC_TAG_REDTRC 0x72545243 /* */
00188 #define JAS_ICC_TAG_SCRNGDES 0x73637264 /* */
00189 #define JAS_ICC_TAG_SCRNG 0x7363726e /* */
00190 #define JAS_ICC_TAG_TECH 0x74656368 /* */
00191 #define JAS_ICC_TAG_UCRBG 0x62666420 /* */
00192 #define JAS_ICC_TAG_VIEWCONDDDESC 0x76756564 /* */
00193 #define JAS_ICC_TAG_VIEWCOND 0x76696577 /* */
00194
00195 /* Type signatures. */
00196 #define JAS_ICC_TYPE_CRDINFO 0x63726469 /* CRD information */
00197 #define JAS_ICC_TYPE_CURV 0x63757276 /* curve */
00198 #define JAS_ICC_TYPE_DATA 0x64617461 /* data */
00199 #define JAS_ICC_TYPE_TIME 0x6474696d /* date/time */
00200 #define JAS_ICC_TYPE_DEVSET 0x64657673 /* device settings */
00201 #define JAS_ICC_TYPE_LUT16 0x6d667432 /* */
00202 #define JAS_ICC_TYPE_LUT8 0x6d667431 /* */
00203 #define JAS_ICC_TYPE_MEASURE 0x6d656173 /* */
00204 #define JAS_ICC_TYPE_NAMCOLR 0x6e636f6c /* */
00205 #define JAS_ICC_TYPE_NAMCOLR2 0x6e636c32 /* */
00206 #define JAS_ICC_TYPE_PROFSEQDESC 0x70736571 /* profile sequence description */
00207 #define JAS_ICC_TYPE_RESPCURVSET16 0x72637332 /* response curve set 16 */
00208 #define JAS_ICC_TYPE_SF32 0x73663332 /* signed 32-bit fixed-point */
00209 #define JAS_ICC_TYPE_SCRNG 0x7363726e /* screening */
00210 #define JAS_ICC_TYPE_SIG 0x73696720 /* signature */
00211 #define JAS_ICC_TYPE_TXTDESC 0x64657363 /* text description */
00212 #define JAS_ICC_TYPE_TXT 0x74657874 /* text */
00213 #define JAS_ICC_TYPE_UF32 0x75663332 /* unsigned 32-bit fixed-point */
00214 #define JAS_ICC_TYPE_UCRBG 0x62666420 /* */
00215 #define JAS_ICC_TYPE_UI16 0x75693136 /* */
00216 #define JAS_ICC_TYPE_UI32 0x75693332 /* */
00217 #define JAS_ICC_TYPE_UI8 0x75693038 /* */
00218 #define JAS_ICC_TYPE_UI64 0x75693634 /* */
00219 #define JAS_ICC_TYPE_VIEWCOND 0x76696577 /* */
00220 #define JAS_ICC_TYPE_XYZ 0x58595a20 /* XYZ */
00221
00222 typedef uint_fast8_t jas_iccuint8_t;
00223 typedef uint_fast16_t jas_iccuint16_t;
00224 typedef uint_fast32_t jas_iccuint32_t;
00225 typedef int_fast32_t jas_iccsint32_t;
00226 typedef int_fast32_t jas_iccs15fixed16_t;
00227 typedef uint_fast32_t jas_iccu16fixed16_t;
00228 typedef uint_fast64_t jas_iccuint64_t;
00229 typedef uint_fast32_t jas_iccsig_t;
00230
00231 typedef jas_iccsig_t jas_icctagsig_t;
00232 typedef jas_iccsig_t jas_icctagtype_t;
00233 typedef jas_iccsig_t jas_iccattrname_t;
00234
00235 /* Date/time type. */
00236 typedef struct {
00237     jas_iccuint16_t year;
00238     jas_iccuint16_t month;
00239     jas_iccuint16_t day;
00240     jas_iccuint16_t hour;
00241     jas_iccuint16_t min;
00242     jas_iccuint16_t sec;
00243 } jas_icctime_t;
00244
00245 /* XYZ type. */
00246 typedef struct {
00247     jas_iccs15fixed16_t x;
00248     jas_iccs15fixed16_t y;
00249     jas_iccs15fixed16_t z;
00250 } jas_iccxyz_t;
00251
00252 /* Curve type. */
00253 typedef struct {
00254     jas_iccuint32_t numents;
00255     jas_iccuint16_t *ents;
00256 } jas_icccurv_t;
00257
00258 /* Text description type. */
00259 typedef struct {

```

```

00260     jas_iccuint32_t asclen;
00261     char *ascdata; /* ASCII invariant description */
00262     jas_iccuint32_t uclangcode; /* Unicode language code */
00263     jas_iccuint32_t uclen; /* Unicode localizable description count */
00264     jas_uchar *ucdata; /* Unicode localizable description */
00265     jas_iccuint16_t sccode; /* ScriptCode code */
00266     jas_iccuint8_t maclen; /* Localizable Macintosh description count */
00267     jas_uchar macdata[69]; /* Localizable Macintosh description */
00268 } jas_iccetxtdesc_t;
00269
00270 /* Text type. */
00271 typedef struct {
00272     char *string; /* ASCII character string */
00273 } jas_iccetxt_t;
00274
00275 typedef struct {
00276     jas_iccuint8_t numinchans;
00277     jas_iccuint8_t numoutchans;
00278     jas_iccsint32_t e[3][3];
00279     jas_iccuint8_t clutlen;
00280     jas_iccuint8_t *clut;
00281     jas_iccuint16_t numintabents;
00282     jas_iccuint8_t **intabs;
00283     jas_iccuint8_t *intabsbuf;
00284     jas_iccuint16_t numouttabents;
00285     jas_iccuint8_t **outtabs;
00286     jas_iccuint8_t *outtabsbuf;
00287 } jas_icclut8_t;
00288
00289 typedef struct {
00290     jas_iccuint8_t numinchans;
00291     jas_iccuint8_t numoutchans;
00292     jas_iccsint32_t e[3][3];
00293     jas_iccuint8_t clutlen;
00294     jas_iccuint16_t *clut;
00295     jas_iccuint16_t numintabents;
00296     jas_iccuint16_t **intabs;
00297     jas_iccuint16_t *intabsbuf;
00298     jas_iccuint16_t numouttabents;
00299     jas_iccuint16_t **outtabs;
00300     jas_iccuint16_t *outtabsbuf;
00301 } jas_icclut16_t;
00302
00303 struct jas_iccattrval_s;
00304
00305 typedef struct {
00306     void (*destroy)(struct jas_iccattrval_s *);
00307     int (*copy)(struct jas_iccattrval_s *, const struct jas_iccattrval_s *);
00308     int (*input)(struct jas_iccattrval_s *, jas_stream_t *, unsigned);
00309     /*#ifdef JAS_ENABLE_ENCODER
00310     int (*output)(struct jas_iccattrval_s *, jas_stream_t *);
00311     /*#endif
00312     unsigned (*getsize)(const struct jas_iccattrval_s *);
00313     void (*dump)(const struct jas_iccattrval_s *, FILE *);
00314 } jas_iccattrvalops_t;
00315
00316 /* Attribute value type (type and value information). */
00317 typedef struct jas_iccattrval_s {
00318     unsigned refcnt; /* reference count */
00319     jas_iccsig_t type; /* type */
00320     const jas_iccattrvalops_t *ops; /* type-dependent operations */
00321     union {
00322         jas_iccxyz_t xyz;
00323         jas_icccurv_t curv;
00324         jas_iccetxtdesc_t txtdesc;
00325         jas_iccetxt_t txt;
00326         jas_icclut8_t lut8;
00327         jas_icclut16_t lut16;
00328     } data; /* value */
00329 } jas_iccattrval_t;
00330
00331 /* Header type. */
00332 typedef struct {
00333     jas_iccuint32_t size; /* profile size */
00334     jas_iccsig_t cmmtype; /* CMM type signature */
00335     jas_iccuint32_t version; /* profile version */
00336     jas_iccsig_t clas; /* profile/device class signature */
00337     jas_iccsig_t colorspace; /* color space of data */
00338     jas_iccsig_t refcolorspace; /* profile connection space */
00339     jas_icctime_t ctime; /* creation time */
00340     jas_iccsig_t magic; /* profile file signature */

```

```

00341     jas_iccsig_t platform; /* primary platform */
00342     jas_iccuint32_t flags; /* profile flags */
00343     jas_iccsig_t maker; /* device manufacturer signature */
00344     jas_iccsig_t model; /* device model signature */
00345     jas_iccuint64_t attr; /* device setup attributes */
00346     jas_iccsig_t intent; /* rendering intent */
00347     jas_iccxyz_t illum; /* illuminant */
00348     jas_iccsig_t creator; /* profile creator signature */
00349 } jas_icchdr_t;
00350
00351 typedef struct {
00352     jas_iccsig_t name;
00353     jas_iccattrval_t *val;
00354 } jas_iccattr_t;
00355
00356 typedef struct {
00357     unsigned numattrs;
00358     unsigned maxattrs;
00359     jas_iccattr_t *attrs;
00360 } jas_iccattrtab_t;
00361
00362 typedef struct jas_icctagtabent_s {
00363     jas_iccuint32_t tag;
00364     jas_iccuint32_t off;
00365     jas_iccuint32_t len;
00366     void *data;
00367     struct jas_icctagtabent_s *first;
00368 } jas_icctagtabent_t;
00369
00370 typedef struct {
00371     jas_iccuint32_t numents;
00372     jas_icctagtabent_t *ents;
00373 } jas_icctagtab_t;
00374
00375 /* ICC profile type. */
00376 typedef struct {
00377     jas_icchdr_t hdr;
00378     jas_icctagtab_t tagtab;
00379     jas_iccattrtab_t attrtab;
00380 } jas_iccprof_t;
00381
00382 typedef struct {
00383     jas_iccuint32_t type;
00384     jas_iccattrvalops_t ops;
00385 } jas_iccattrvalinfo_t;
00386
00387 JAS_DLLEXPORT jas_iccprof_t *jas_iccprof_load(jas_stream_t *in);
00388 JAS_DLLEXPORT int jas_iccprof_save(jas_iccprof_t *prof, jas_stream_t *out);
00389 JAS_DLLEXPORT void jas_iccprof_destroy(jas_iccprof_t *prof);
00390 JAS_ATTRIBUTE_PURE
00391 JAS_DLLEXPORT jas_iccattrval_t *jas_iccprof_getattr(const jas_iccprof_t *prof,
00392     jas_iccattrname_t name);
00393 JAS_DLLEXPORT int jas_iccprof_setattr(jas_iccprof_t *prof, jas_iccattrname_t name,
00394     jas_iccattrval_t *val);
00395 JAS_DLLEXPORT void jas_iccprof_dump(const jas_iccprof_t *prof, FILE *out);
00396 JAS_DLLEXPORT jas_iccprof_t *jas_iccprof_copy(const jas_iccprof_t *prof);
00397 JAS_DLLEXPORT int jas_iccprof_gethdr(const jas_iccprof_t *prof, jas_icchdr_t *hdr);
00398 JAS_DLLEXPORT int jas_iccprof_sethdr(jas_iccprof_t *prof, const jas_icchdr_t *hdr);
00399
00400 JAS_DLLEXPORT void jas_iccattrval_destroy(jas_iccattrval_t *attrval);
00401 JAS_DLLEXPORT void jas_iccattrval_dump(const jas_iccattrval_t *attrval, FILE *out);
00402 JAS_DLLEXPORT int jas_iccattrval_allowmodify(jas_iccattrval_t **attrval);
00403 JAS_DLLEXPORT jas_iccattrval_t *jas_iccattrval_clone(jas_iccattrval_t *attrval);
00404 JAS_DLLEXPORT jas_iccattrval_t *jas_iccattrval_create(jas_iccuint32_t type);
00405
00406 JAS_DLLEXPORT void jas_iccattrtab_dump(const jas_iccattrtab_t *attrtab, FILE *out);
00407
00408 JAS_DLLEXPORT extern const jas_uchar jas_iccprofdata_srgb[];
00409 JAS_DLLEXPORT extern const unsigned jas_iccprofdata_srgbllen;
00410 JAS_DLLEXPORT extern const jas_uchar jas_iccprofdata_sgray[];
00411 JAS_DLLEXPORT extern const unsigned jas_iccprofdata_sgraylen;
00412 JAS_DLLEXPORT jas_iccprof_t *jas_iccprof_createfrombuf(const jas_uchar *buf, unsigned len);
00413 JAS_DLLEXPORT jas_iccprof_t *jas_iccprof_createfromclrspc(unsigned clrspc);
00414
00415 #ifdef __cplusplus
00416 }
00417 #endif
00418
00419 #endif

```

## 11.17 jas\_image.h File Reference

JasPer Image Class.

```
#include <jasper/jas_config.h>
#include <jasper/jas_stream.h>
#include <jasper/jas_types.h>
#include <jasper/jas_seq.h>
#include <jasper/jas_cm.h>
#include <stdio.h>
```

### Classes

- struct [jas\\_image\\_cmpt\\_t](#)  
*Image component class.*
- struct [jas\\_image\\_t](#)  
*Image class.*
- struct [jas\\_image\\_cmptparm\\_t](#)  
*Component parameters class.*
- struct [jas\\_image\\_fmtops\\_t](#)  
*Image format-dependent operations.*
- struct [jas\\_image\\_fmtinfo\\_t](#)  
*Image format information.*

### Macros

- #define [JAS\\_IMAGE\\_MAXFMTS](#) 32  
*The maximum number of image data formats supported.*
- #define [jas\\_image\\_width](#)(image) ((image)->brx\_ - (image)->tlx\_)  
*Get the width of the image in units of the image reference grid.*
- #define [jas\\_image\\_height](#)(image) ((image)->bry\_ - (image)->tly\_)  
*Get the height of the image in units of the image reference grid.*
- #define [jas\\_image\\_tlx](#)(image) ((image)->tlx\_)  
*Get the x-coordinate of the top-left corner of the image bounding box on the reference grid.*
- #define [jas\\_image\\_tly](#)(image) ((image)->tly\_)  
*Get the y-coordinate of the top-left corner of the image bounding box on the reference grid.*
- #define [jas\\_image\\_brx](#)(image) ((image)->brx\_)  
*Get the x-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).*
- #define [jas\\_image\\_bry](#)(image) ((image)->bry\_)  
*Get the y-coordinate of the bottom-right corner of the image bounding box on the reference grid (plus one).*
- #define [jas\\_image\\_numcmpts](#)(image) ((image)->numcmpts\_)  
*Get the number of image components.*
- #define [jas\\_image\\_clrspc](#)(image) ((image)->clrspc\_)  
*Get the color model used by the image.*
- #define [jas\\_image\\_setclrspc](#)(image, clrspc) ((image)->clrspc\_ = (clrspc))

- *Set the color model for an image.*
- `#define jas_image_cmptwidth(image, cmptno) ((image)->cmpts_[cmptno]->width_)`  
*Get the width of a component.*
- `#define jas_image_cmptheight(image, cmptno) ((image)->cmpts_[cmptno]->height_)`  
*Get the height of a component.*
- `#define jas_image_cmptsgnd(image, cmptno) ((image)->cmpts_[cmptno]->sgnd_)`  
*Get the signedness of the sample data for a component.*
- `#define jas_image_cmptprec(image, cmptno) ((image)->cmpts_[cmptno]->prec_)`  
*Get the precision of the sample data for a component.*
- `#define jas_image_cmptstep(image, cmptno) ((image)->cmpts_[cmptno]->hstep_)`  
*Get the horizontal subsampling factor for a component.*
- `#define jas_image_cmptvstep(image, cmptno) ((image)->cmpts_[cmptno]->vstep_)`  
*Get the vertical subsampling factor for a component.*
- `#define jas_image_cmpttlx(image, cmptno) ((image)->cmpts_[cmptno]->tlx_)`  
*Get the x-coordinate of the top-left corner of a component.*
- `#define jas_image_cmpttly(image, cmptno) ((image)->cmpts_[cmptno]->tly_)`  
*Get the y-coordinate of the top-left corner of a component.*
- `#define jas_image_cmptbrx(image, cmptno)`  
*Get the x-coordinate of the bottom-right corner of a component (plus "one").*
- `#define jas_image_cmptbry(image, cmptno)`  
*Get the y-coordinate of the bottom-right corner of a component (plus "one").*
- `#define jas_image_cmprofn(image) ((image)->cmprofn_)`  
*Get the color management profile of an image.*
- `#define jas_image_setcmprofn(image, cmprofn) ((image)->cmprofn_ = cmprofn)`  
*Set the color management profile for an image.*

## Typedefs

- `typedef int_fast32_t jas_image_coord_t`  
*Image coordinate.*
- `typedef int_fast16_t jas_image_colorspc_t`  
*Color space (e.g., RGB, YCbCr).*
- `typedef int_fast32_t jas_image_cmpttype_t`  
*Component type (e.g., color, opacity).*
- `typedef int_fast16_t jas_image_smpltype_t`  
*Component sample data format (e.g., real/integer, signedness, precision).*

## Functions

- `JAS_DLLEXPORT jas_image_t * jas_image_create (unsigned numcmpts, const jas_image_cmptparm_t *cmptparms, jas_clrspc_t clrspc)`  
*Create an image.*
- `JAS_DLLEXPORT jas_image_t * jas_image_create0 (void)`  
*Create an "empty" image.*
- `JAS_DLLEXPORT jas_image_t * jas_image_copy (jas_image_t *image)`  
*Clone an image.*

- JAS\_DLLEXPORT void [jas\\_image\\_destroy](#) ([jas\\_image\\_t](#) \*image)  
*Deallocate any resources associated with an image.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT bool [jas\\_image\\_cmpt\\_domains\\_same](#) (const [jas\\_image\\_t](#) \*image)  
*Test if all components are specified at the same positions in space.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT uint\_fast32\_t [jas\\_image\\_rawsize](#) (const [jas\\_image\\_t](#) \*image)  
*Get the raw size of an image (i.e., the nominal size of the image without any compression).*
- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_decode](#) ([jas\\_stream\\_t](#) \*in, int fmt, const char \*optstr)  
*Create an image from a stream in some specified format.*
- JAS\_DLLEXPORT int [jas\\_image\\_encode](#) ([jas\\_image\\_t](#) \*image, [jas\\_stream\\_t](#) \*out, int fmt, const char \*optstr)  
*Write an image to a stream in a specified format.*
- JAS\_DLLEXPORT int [jas\\_image\\_readcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, [jas\\_matrix\\_t](#) \*data)  
*Read a rectangular region of an image component.*
- JAS\_DLLEXPORT int [jas\\_image\\_writecmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, const [jas\\_matrix\\_t](#) \*data)  
*Write a rectangular region of an image component.*
- JAS\_DLLEXPORT void [jas\\_image\\_delcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno)  
*Delete a component from an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_addcmpt](#) ([jas\\_image\\_t](#) \*image, int cmptno, const [jas\\_image\\_cmptparm\\_t](#) \*cmptparm)  
*Add a component to an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_copycmpt](#) ([jas\\_image\\_t](#) \*dstimage, unsigned dstcmptno, [jas\\_image\\_t](#) \*srcimage, unsigned srccmptno)  
*Copy a component from one image to another.*
- JAS\_DLLEXPORT int [jas\\_image\\_depalettize](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned numlutents, const int\_fast32\_t \*lutents, unsigned dtype, unsigned newcmptno)  
*Depalettize an image.*
- JAS\_DLLEXPORT int [jas\\_image\\_readcmptsample](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned x, unsigned y)  
*Read a component sample for an image.*
- JAS\_DLLEXPORT void [jas\\_image\\_writecmptsample](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned x, unsigned y, int\_fast32\_t v)  
*Write a component sample for an image.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_getcmptbytype](#) (const [jas\\_image\\_t](#) \*image, [jas\\_image\\_cmpttype\\_t](#) ctype)  
*Get an image component by its type.*
- JAS\_DLLEXPORT void [jas\\_image\\_clearfmts](#) (void)  
*Clear the table of image formats.*
- JAS\_DLLEXPORT int [jas\\_image\\_addfmt](#) (int id, const char \*name, const char \*ext, const char \*desc, const [jas\\_image\\_fmtops\\_t](#) \*ops)  
*Add entry to table of image formats.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_strtofmt](#) (const char \*s)  
*Get the ID for the image format with the specified name.*
- JAS\_ATTRIBUTE\_CONST JAS\_DLLEXPORT const char \* [jas\\_image\\_fmtostr](#) (int fmt)  
*Get the name of the image format with the specified ID.*
- JAS\_ATTRIBUTE\_CONST JAS\_DLLEXPORT const [jas\\_image\\_fmtinfo\\_t](#) \* [jas\\_image\\_lookupfmtbyid](#) (int id)  
*Lookup image format information by the format ID.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT const [jas\\_image\\_fmtinfo\\_t](#) \* [jas\\_image\\_lookupfmtbyname](#) (const char \*name)

*Lookup image format information by the format name.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_fmtfromname](#) (const char \*filename)

*Guess the format of an image file based on its name.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_getfmt](#) (jas\_stream\_t \*in)

*Get the format of image data in a stream.*

- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_image\\_ishomosamp](#) (const [jas\\_image\\_t](#) \*image)

???

- JAS\_DLLEXPORT int [jas\\_image\\_sampcmpt](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, unsigned newcmptno, [jas\\_image\\_coord\\_t](#) ho, [jas\\_image\\_coord\\_t](#) vo, [jas\\_image\\_coord\\_t](#) hs, [jas\\_image\\_coord\\_t](#) vs, int sgnd, unsigned prec)

???

- JAS\_DLLEXPORT int [jas\\_image\\_writecmpt2](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, const long \*buf)

*Write sample data in a component of an image.*

- JAS\_DLLEXPORT int [jas\\_image\\_readcmpt2](#) ([jas\\_image\\_t](#) \*image, unsigned cmptno, [jas\\_image\\_coord\\_t](#) x, [jas\\_image\\_coord\\_t](#) y, [jas\\_image\\_coord\\_t](#) width, [jas\\_image\\_coord\\_t](#) height, long \*buf)

*Read sample data in a component of an image.*

- JAS\_DLLEXPORT [jas\\_image\\_t](#) \* [jas\\_image\\_chclrspc](#) ([jas\\_image\\_t](#) \*image, const [jas\\_cmprof\\_t](#) \*outprof, [jas\\_cmxfm\\_intent\\_t](#) intent)

*Change the color space for an image.*

- JAS\_DLLEXPORT void [jas\\_image\\_dump](#) ([jas\\_image\\_t](#) \*image, FILE \*out)

*Dump the information for an image (for debugging).*

### 11.17.1 Detailed Description

JasPer Image Class.

## 11.18 jas\_image.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
```

```

00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00069 #ifndef JAS_IMAGE_H
00070 #define JAS_IMAGE_H
00071
00072 /*****
00073 * Includes.
00074 *****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h>
00078
00079 #include <jasper/jas_stream.h>
00080 #include <jasper/jas_types.h>
00081 #include <jasper/jas_seq.h> /* IWYU pragma: export */
00082 #include <jasper/jas_cm.h> /* IWYU pragma: export */
00083 #include <stdio.h>
00084
00085 #ifdef __cplusplus
00086 extern "C" {
00087 #endif
00088
00089 /*****
00090 * Constants.
00091 *****/
00092
00093 /*
00094 * Miscellaneous constants.
00095 */
00096
00097 /* Basic units */
00098 #define JAS_IMAGE_KIBI (JAS_CAST(size_t, 1024))
00099 #define JAS_IMAGE_MEBI (JAS_IMAGE_KIBI * JAS_IMAGE_KIBI)
00100
00101 /* The threshold at which image data is no longer stored in memory. */
00102 #define JAS_IMAGE_INMEMTHRESH (256 * JAS_IMAGE_MEBI)
00103
00104 /*
00105 * Component types
00106 */
00107
00108 #define JAS_IMAGE_CT_UNKNOWN 0x10000
00109 #define JAS_IMAGE_CT_COLOR(n) ((n) & 0x7fff)
00110 #define JAS_IMAGE_CT_OPACITY 0x08000
00111
00112 #define JAS_IMAGE_CT_RGB_R 0

```



```

00118 #define JAS_IMAGE_CT_RGB_G      1
00119 #define JAS_IMAGE_CT_RGB_B      2
00120
00121 #define JAS_IMAGE_CT_YCBCR_Y    0
00122 #define JAS_IMAGE_CT_YCBCR_CB   1
00123 #define JAS_IMAGE_CT_YCBCR_CR   2
00124
00125 #define JAS_IMAGE_CT_GRAY_Y     0
00126
00127 /*****
00128  * Simple types.
00129  *****/
00130
00133 typedef int_fast32_t jas_image_coord_t;
00134 #define JAS_IMAGE_COORD_MAX INT_FAST32_MAX
00135 #define JAS_IMAGE_COORD_MIN INT_FAST32_MIN
00136
00140 typedef int_fast16_t jas_image_colorspc_t;
00141
00145 typedef int_fast32_t jas_image_cmpttype_t;
00146
00151 typedef int_fast16_t jas_image_smpltype_t;
00152
00153 /*****
00154  * Image class and supporting classes.
00155  *****/
00156
00160 typedef struct {
00161
00162     jas_image_coord_t tlx_;
00163     /* The x-coordinate of the top-left corner of the component. */
00164
00165     jas_image_coord_t tly_;
00166     /* The y-coordinate of the top-left corner of the component. */
00167
00168     jas_image_coord_t hstep_;
00169     /* The horizontal sampling period in units of the reference grid. */
00170
00171     jas_image_coord_t vstep_;
00172     /* The vertical sampling period in units of the reference grid. */
00173
00174     jas_image_coord_t width_;
00175     /* The component width in samples. */
00176
00177     jas_image_coord_t height_;
00178     /* The component height in samples. */
00179
00180     unsigned prec_;
00181     /* The precision of the sample data (i.e., the number of bits per
00182     sample). If the samples are signed values, this quantity
00183     includes the sign bit. */
00184
00185     int sgnd_;
00186     /* The signedness of the sample data. */
00187
00188     jas_stream_t *stream_;
00189     /* The stream containing the component data. */
00190
00191     unsigned cps_;
00192     /* The number of characters per sample in the stream. */
00193
00194     jas_image_cmpttype_t type_;
00195     /* The type of component (e.g., opacity, red, green, blue, luma). */
00196
00197 } jas_image_cmpt_t;
00198
00202 typedef struct {
00203
00204     jas_image_coord_t tlx_;
00205     /* The x-coordinate of the top-left corner of the image bounding box. */
00206
00207     jas_image_coord_t tly_;
00208     /* The y-coordinate of the top-left corner of the image bounding box. */
00209
00210     jas_image_coord_t brx_;
00211     /* The x-coordinate of the bottom-right corner of the image bounding
00212     box (plus one). */
00213
00214     jas_image_coord_t bry_;
00215     /* The y-coordinate of the bottom-right corner of the image bounding
00216     box (plus one). */

```

```

00217
00218     unsigned numcmpts_;
00219     /* The number of components. */
00220
00221     unsigned maxcmpts_;
00222     /* The maximum number of components that this image can have (i.e., the
00223        allocated size of the components array). */
00224
00225     jas_image_cmpt_t **cmpts_;
00226     /* Per-component information. */
00227
00228     jas_clrspc_t clrspc_;
00229
00230     jas_cmpprof_t *cmpprof_;
00231
00232     // bool inmem_;
00233
00234 } jas_image_t;
00235
00243 typedef struct {
00244
00245     jas_image_coord_t tlx;
00246     /* The x-coordinate of the top-left corner of the component. */
00247
00248     jas_image_coord_t tly;
00249     /* The y-coordinate of the top-left corner of the component. */
00250
00251     jas_image_coord_t hstep;
00252     /* The horizontal sampling period in units of the reference grid. */
00253
00254     jas_image_coord_t vstep;
00255     /* The vertical sampling period in units of the reference grid. */
00256
00257     jas_image_coord_t width;
00258     /* The width of the component in samples. */
00259
00260     jas_image_coord_t height;
00261     /* The height of the component in samples. */
00262
00263     unsigned prec;
00264     /* The precision of the component sample data. */
00265
00266     int sgnd;
00267     /* The signedness of the component sample data. */
00268
00269 } jas_image_cmptparm_t;
00270
00271 /*****
00272  * File format related classes.
00273  \*****/
00274
00278 #define JAS_IMAGE_MAXFMTS      32
00279
00283 typedef struct {
00284
00285     jas_image_t *(*decode)(jas_stream_t *in, const char *opts);
00286     /* Decode image data from a stream. */
00287
00288     int (*encode)(jas_image_t *image, jas_stream_t *out, const char *opts);
00289     /* Encode image data to a stream. */
00290
00291     int (*validate)(jas_stream_t *in);
00292     /* Determine if stream data is in a particular format. */
00293
00294 } jas_image_fmtops_t;
00295
00299 typedef struct {
00300
00301     int id;
00302     /* The ID for this format. */
00303
00304     char *name;
00305     /* The name by which this format is identified. */
00306
00307     char *ext;
00308     /* The file name extension associated with this format. */
00309
00310     char *desc;
00311     /* A brief description of the format. */
00312
00313     jas_image_fmtops_t ops;

```

```

00314      /* The operations for this format. */
00315
00316 } jas_image_fmtinfo_t;
00317
00318 /*****
00319  * Image operations.
00320  *****/
00321
00325 JAS_DLLEXPORT jas_image_t *jas_image_create(unsigned numcmpts,
00326      const jas_image_cmptparm_t *cmptparms, jas_clrspc_t clrspc);
00327
00331 JAS_DLLEXPORT jas_image_t *jas_image_create0(void);
00332
00336 JAS_DLLEXPORT jas_image_t *jas_image_copy(jas_image_t *image);
00337
00341 JAS_DLLEXPORT void jas_image_destroy(jas_image_t *image);
00342
00346 #define jas_image_width(image) \
00347     ((image)->brx_ - (image)->tlx_)
00348
00352 #define jas_image_height(image) \
00353     ((image)->bry_ - (image)->tly_)
00354
00359 #define jas_image_tlx(image) \
00360     ((image)->tlx_)
00361
00366 #define jas_image_tly(image) \
00367     ((image)->tly_)
00368
00373 #define jas_image_brx(image) \
00374     ((image)->brx_)
00375
00380 #define jas_image_bry(image) \
00381     ((image)->bry_)
00382
00386 #define jas_image_numcmpts(image) \
00387     ((image)->numcmpts_)
00388
00392 #define jas_image_clrspc(image) \
00393     ((image)->clrspc_)
00394
00398 #define jas_image_setclrspc(image, clrspc) \
00399     ((image)->clrspc_ = (clrspc))
00400
00401 #define jas_image_cmpttype(image, cmptno) \
00402     ((image)->cmpts_[cmptno]->type_)
00403 #define jas_image_setcmpttype(image, cmptno, type) \
00404     ((image)->cmpts_[cmptno]->type_ = (type))
00405
00409 #define jas_image_cmptwidth(image, cmptno) \
00410     ((image)->cmpts_[cmptno]->width_)
00411
00415 #define jas_image_cmptheight(image, cmptno) \
00416     ((image)->cmpts_[cmptno]->height_)
00417
00421 #define jas_image_cmptsgnd(image, cmptno) \
00422     ((image)->cmpts_[cmptno]->sgnd_)
00423
00427 #define jas_image_cmptprec(image, cmptno) \
00428     ((image)->cmpts_[cmptno]->prec_)
00429
00433 #define jas_image_cmptstep(image, cmptno) \
00434     ((image)->cmpts_[cmptno]->hstep_)
00435
00439 #define jas_image_cmptvstep(image, cmptno) \
00440     ((image)->cmpts_[cmptno]->vstep_)
00441
00445 #define jas_image_cmpttlx(image, cmptno) \
00446     ((image)->cmpts_[cmptno]->tlx_)
00447
00451 #define jas_image_cmpttly(image, cmptno) \
00452     ((image)->cmpts_[cmptno]->tly_)
00453
00458 #define jas_image_cmptbrx(image, cmptno) \
00459     ((image)->cmpts_[cmptno]->tlx_ + (image)->cmpts_[cmptno]->width_ * \
00460     (image)->cmpts_[cmptno]->hstep_)
00461
00466 #define jas_image_cmptbry(image, cmptno) \
00467     ((image)->cmpts_[cmptno]->tly_ + (image)->cmpts_[cmptno]->height_ * \
00468     (image)->cmpts_[cmptno]->vstep_)
00469

```

```

00473 JAS_ATTRIBUTE_PURE
00474 JAS_DLLEXPORT bool jas_image_cmpt_domains_same(const jas_image_t *image);
00475
00480 JAS_ATTRIBUTE_PURE
00481 JAS_DLLEXPORT
00482 uint_fast32_t jas_image_rawsize(const jas_image_t *image);
00483
00487 JAS_DLLEXPORT
00488 jas_image_t *jas_image_decode(jas_stream_t *in, int fmt, const char *optstr);
00489
00493 JAS_DLLEXPORT
00494 int jas_image_encode(jas_image_t *image, jas_stream_t *out, int fmt,
00495     const char *optstr);
00496
00504 JAS_DLLEXPORT
00505 int jas_image_readcmpt(jas_image_t *image, unsigned cmptno,
00506     jas_image_coord_t x, jas_image_coord_t y, jas_image_coord_t width,
00507     jas_image_coord_t height, jas_matrix_t *data);
00508
00512 JAS_DLLEXPORT
00513 int jas_image_writecmpt(jas_image_t *image, unsigned cmptno,
00514     jas_image_coord_t x, jas_image_coord_t y, jas_image_coord_t width,
00515     jas_image_coord_t height, const jas_matrix_t *data);
00516
00520 JAS_DLLEXPORT void jas_image_delcmpt(jas_image_t *image, unsigned cmptno);
00521
00525 JAS_DLLEXPORT int jas_image_addcmpt(jas_image_t *image, int cmptno,
00526     const jas_image_cmptparm_t *cmptparm);
00527
00531 JAS_DLLEXPORT int jas_image_copycmpt(jas_image_t *dstimage, unsigned dstcmptno,
00532     jas_image_t *srcimage, unsigned srccmptno);
00533
00534 JAS_ATTRIBUTE_CONST
00535 static inline bool JAS_IMAGE_CDT_GETSGND(uint_least8_t dtype)
00536 {
00537     return (dtype >> 7) & 1;
00538 }
00539
00540 JAS_ATTRIBUTE_CONST
00541 static inline uint_least8_t JAS_IMAGE_CDT_SETSGND(bool sgnd)
00542 {
00543     return (uint_least8_t)sgnd << 7;
00544 }
00545
00546 JAS_ATTRIBUTE_CONST
00547 static inline uint_least8_t JAS_IMAGE_CDT_GETPREC(uint_least8_t dtype)
00548 {
00549     return dtype & 0x7f;
00550 }
00551
00552 JAS_ATTRIBUTE_CONST
00553 static inline uint_least8_t JAS_IMAGE_CDT_SETPREC(uint_least8_t dtype)
00554 {
00555     return dtype & 0x7f;
00556 }
00557
00558 JAS_ATTRIBUTE_PURE
00559 static inline uint_least8_t jas_image_cmptdtype(const jas_image_t *image, unsigned cmptno)
00560 {
00561     return JAS_IMAGE_CDT_SETSGND(image->cmpts_[cmptno]->sgnd_) |
00562         JAS_IMAGE_CDT_SETPREC(image->cmpts_[cmptno]->prec_);
00563 }
00564
00568 JAS_DLLEXPORT
00569 int jas_image_depalettize(jas_image_t *image, unsigned cmptno,
00570     unsigned numlutents, const int_fast32_t *lutents, unsigned dtype,
00571     unsigned newcmptno);
00572
00576 JAS_DLLEXPORT
00577 int jas_image_readcmptsample(jas_image_t *image, unsigned cmptno, unsigned x,
00578     unsigned y);
00579
00583 JAS_DLLEXPORT
00584 void jas_image_writecmptsample(jas_image_t *image, unsigned cmptno,
00585     unsigned x, unsigned y, int_fast32_t v);
00586
00590 JAS_ATTRIBUTE_PURE
00591 JAS_DLLEXPORT
00592 int jas_image_getcmptbytype(const jas_image_t *image, jas_image_cmpttype_t ctype);
00593
00594 /*****

```

```

00595 * Image format-related operations.
00596 \*****
00597
00601 JAS_DLLEXPORT
00602 void jas_image_clearfmts(void);
00603
00607 JAS_DLLEXPORT
00608 int jas_image_addfmt(int id, const char *name, const char *ext,
00609     const char *desc, const jas_image_fmtops_t *ops);
00610
00614 JAS_ATTRIBUTE_PURE
00615 JAS_DLLEXPORT
00616 int jas_image_strtofmt(const char *s);
00617
00621 JAS_ATTRIBUTE_CONST
00622 JAS_DLLEXPORT
00623 const char *jas_image_fmtostr(int fmt);
00624
00628 JAS_ATTRIBUTE_CONST
00629 JAS_DLLEXPORT
00630 const jas_image_fmtinfo_t *jas_image_lookupfmtbyid(int id);
00631
00635 JAS_ATTRIBUTE_PURE
00636 JAS_DLLEXPORT
00637 const jas_image_fmtinfo_t *jas_image_lookupfmtbyname(const char *name);
00638
00642 JAS_ATTRIBUTE_PURE
00643 JAS_DLLEXPORT
00644 int jas_image_fmtfromname(const char *filename);
00645
00649 JAS_ATTRIBUTE_PURE
00650 JAS_DLLEXPORT
00651 int jas_image_getfmt(jas_stream_t *in);
00652
00653
00657 #define jas_image_cmprof(image) ((image)->cmprof_)
00658
00662 JAS_ATTRIBUTE_PURE
00663 JAS_DLLEXPORT
00664 int jas_image_ishomosamp(const jas_image_t *image);
00665
00669 JAS_DLLEXPORT
00670 int jas_image_sampcmpt(jas_image_t *image, unsigned cmptno, unsigned newcmptno,
00671     jas_image_coord_t ho, jas_image_coord_t vo, jas_image_coord_t hs,
00672     jas_image_coord_t vs, int sgnd, unsigned prec);
00673
00677 JAS_DLLEXPORT int jas_image_writecmpt2(jas_image_t *image, unsigned cmptno, jas_image_coord_t x,
00678     jas_image_coord_t y, jas_image_coord_t width, jas_image_coord_t height,
00679     const long *buf);
00680
00684 JAS_DLLEXPORT
00685 int jas_image_readcmpt2(jas_image_t *image, unsigned cmptno,
00686     jas_image_coord_t x, jas_image_coord_t y, jas_image_coord_t width,
00687     jas_image_coord_t height, long *buf);
00688
00692 #define jas_image_setcmprof(image, cmprof) ((image)->cmprof_ = cmprof)
00693
00697 JAS_DLLEXPORT jas_image_t *jas_image_chclrspc(jas_image_t *image, const jas_cmprof_t *outprof,
00698     jas_cmxform_intent_t intent);
00699
00703 JAS_DLLEXPORT void jas_image_dump(jas_image_t *image, FILE *out);
00704
00705 /*****
00706 * Image format-dependent operations.
00707 \*****
00708
00709 #if defined(JAS_INCLUDE_JPG_CODEC)
00710 /* Format-dependent operations for JPG support. */
00711 JAS_DLLEXPORT jas_image_t *jpg_decode(jas_stream_t *in, const char *optstr);
00712 JAS_DLLEXPORT int jpg_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00713 JAS_DLLEXPORT int jpg_validate(jas_stream_t *in);
00714 #endif
00715
00716 #if defined(JAS_INCLUDE_MIF_CODEC)
00717 /* Format-dependent operations for MIF support. */
00718 JAS_DLLEXPORT jas_image_t *mif_decode(jas_stream_t *in, const char *optstr);
00719 JAS_DLLEXPORT int mif_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00720 JAS_DLLEXPORT int mif_validate(jas_stream_t *in);
00721 #endif
00722
00723 #if defined(JAS_INCLUDE_PNM_CODEC)

```

```

00724 /* Format-dependent operations for PNM support. */
00725 JAS_DLLEXPORT jas_image_t *pnm_decode(jas_stream_t *in, const char *optstr);
00726 JAS_DLLEXPORT int pnm_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00727 JAS_DLLEXPORT int pnm_validate(jas_stream_t *in);
00728 #endif
00729
00730 #if defined(JAS_INCLUDE_RAS_CODEC)
00731 /* Format-dependent operations for Sun Rasterfile support. */
00732 JAS_DLLEXPORT jas_image_t *ras_decode(jas_stream_t *in, const char *optstr);
00733 JAS_DLLEXPORT int ras_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00734 JAS_DLLEXPORT int ras_validate(jas_stream_t *in);
00735 #endif
00736
00737 #if defined(JAS_INCLUDE_BMP_CODEC)
00738 /* Format-dependent operations for BMP support. */
00739 JAS_DLLEXPORT jas_image_t *bmp_decode(jas_stream_t *in, const char *optstr);
00740 JAS_DLLEXPORT int bmp_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00741 JAS_DLLEXPORT int bmp_validate(jas_stream_t *in);
00742 #endif
00743
00744 #if defined(JAS_INCLUDE_JP2_CODEC)
00745 /* Format-dependent operations for JP2 support. */
00746 JAS_DLLEXPORT jas_image_t *jp2_decode(jas_stream_t *in, const char *optstr);
00747 JAS_DLLEXPORT int jp2_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00748 JAS_DLLEXPORT int jp2_validate(jas_stream_t *in);
00749 #endif
00750
00751 #if defined(JAS_INCLUDE_JPC_CODEC)
00752 /* Format-dependent operations for JPEG-2000 code stream support. */
00753 JAS_DLLEXPORT jas_image_t *jpc_decode(jas_stream_t *in, const char *optstr);
00754 JAS_DLLEXPORT int jpc_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00755 JAS_DLLEXPORT int jpc_validate(jas_stream_t *in);
00756 #endif
00757
00758 #if defined(JAS_INCLUDE_PGX_CODEC)
00759 /* Format-dependent operations for PGX support. */
00760 JAS_DLLEXPORT jas_image_t *pgx_decode(jas_stream_t *in, const char *optstr);
00761 JAS_DLLEXPORT int pgx_encode(jas_image_t *image, jas_stream_t *out, const char *optstr);
00762 JAS_DLLEXPORT int pgx_validate(jas_stream_t *in);
00763 #endif
00764
00769 #ifdef __cplusplus
00770 }
00771 #endif
00772
00773 #endif

```

## 11.19 jas\_init.h File Reference

JasPer Initialization/Cleanup Code.

```
#include <jasper/jas_config.h>
```

### Functions

- JAS\_DLLEXPORT int [jas\\_init](#) (void)  
*Initialize the JasPer library.*
- JAS\_DLLEXPORT void [jas\\_cleanup](#) (void)  
*Perform any clean up for the JasPer library.*

### 11.19.1 Detailed Description

JasPer Initialization/Cleanup Code.

## 11.20 jas\_init.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef JAS_INIT_H
00063 #define JAS_INIT_H
00064
00065
00066 /* The configuration header file should be included first. */
00067 #include <jasper/jas_config.h>
00068
00069 #ifdef __cplusplus
00070 extern "C" {
00071 #endif
00072
00073 /*****
00074  * Functions.
00075  *****/
00076
00077 JAS_DLLEXPORT

```

```

00098 int  jas_init(void);
00099
00106 JAS_DLLEXPORT
00107 void jas_cleanup(void);
00108
00113 #ifdef __cplusplus
00114 }
00115 #endif
00116
00117 #endif

```

## 11.21 jas\_malloc.h File Reference

JasPer Memory Allocator.

```

#include <jasper/jas_config.h>
#include <stdio.h>

```

### 11.21.1 Detailed Description

JasPer Memory Allocator.

## 11.22 jas\_malloc.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS

```



```

00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.  IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.  NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE.  AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY.  THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES").  THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064
00070 #ifndef JAS_MALLOC_H
00071 #define JAS_MALLOC_H
00072
00073 /*****\
00074  * Includes.
00075 \*****/
00076
00077 /* The configuration header file should be included first. */
00078 #include <jasper/jas_config.h>
00079
00080 #include <stdio.h>
00081
00082 #ifdef __cplusplus
00083 extern "C" {
00084 #endif
00085
00086 /*****\
00087  * Functions.
00088 \*****/
00089
00090 /* Allocate memory. */
00091 JAS_DLLEXPORT void *jas_malloc(size_t size);
00092
00093 /* Free memory. */
00094 JAS_DLLEXPORT void jas_free(void *ptr);
00095
00096 /* Resize a block of allocated memory. */
00097 JAS_DLLEXPORT void *jas_realloc(void *ptr, size_t size);
00098
00099 /* Allocate a block of memory and initialize the contents to zero. */
00100 JAS_DLLEXPORT void *jas_calloc(size_t num_elements, size_t element_size);
00101
00102 /* Allocate array (with overflow checking) . */
00103 JAS_DLLEXPORT void *jas_alloc2(size_t num_elements, size_t element_size);
00104
00105 /* Allocate array of arrays (with overflow checking) . */
00106 JAS_DLLEXPORT void *jas_alloc3(size_t num_arrays, size_t array_size, size_t element_size);
00107
00108 /* Resize a block of allocated memory (with overflow checking) . */
00109 JAS_DLLEXPORT void *jas_realloc2(void *ptr, size_t num_elements, size_t element_size);
00110
00111 #if defined(JAS_DEFAULT_MAX_MEM_USAGE)
00112
00113 JAS_DLLEXPORT void jas_set_max_mem_usage(size_t max_mem);
00114
00115 JAS_ATTRIBUTE_PURE
00116 JAS_DLLEXPORT size_t jas_get_mem_usage();
00117
00118 #endif
00119
00120 #ifdef __cplusplus
00121 }
00122 #endif

```

```
00123
00124 #endif
```

## 11.23 jas\_math.h File Reference

Math-Related Code.

```
#include <jasper/jas_config.h>
#include <jasper/jas_compiler.h>
#include <jasper/jas_types.h>
#include <assert.h>
#include <string.h>
#include <stdint.h>
#include <limits.h>
```

### 11.23.1 Detailed Description

Math-Related Code.

## 11.24 jas\_math.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
```

```

00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00069 #ifndef JAS_MATH_H
00070 #define JAS_MATH_H
00071
00072 /*****\
00073 * Includes
00074 \*****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h>
00078
00079 #include <jasper/jas_compiler.h>
00080 #include <jasper/jas_types.h>
00081
00082 #include <assert.h>
00083 #include <string.h>
00084 #include <stdint.h>
00085 #include <limits.h>
00086
00087 #ifdef __cplusplus
00088 extern "C" {
00089 #endif
00090
00091 /*****\
00092 * Macros
00093 \*****/
00094
00095 #define JAS_KIBI          JAS_CAST(size_t, 1024)
00096 #define JAS_MEBI          (JAS_KIBI * JAS_KIBI)
00097
00098 /* Compute the absolute value. */
00099 #define JAS_ABS(x) \
00100     ((x) >= 0) ? (x) : (-x))
00101
00102 /* Compute the minimum of two values. */
00103 #define JAS_MIN(x, y) \
00104     ((x) < (y)) ? (x) : (y)
00105
00106 /* Compute the maximum of two values. */
00107 #define JAS_MAX(x, y) \
00108     ((x) > (y)) ? (x) : (y)
00109
00110 /* Compute the remainder from division (where division is defined such
00111    that the remainder is always nonnegative). */
00112 #define JAS_MOD(x, y) \
00113     (((x) < 0) ? (((-x) % (y)) ? ((y) - ((-x) % (y))) : (0)) : ((x) % (y)))
00114
00115 /* Compute the integer with the specified number of least significant bits
00116    set to one. */
00117 #define JAS_ONES(n) \
00118     ((1 « (n)) - 1)
00119
00120 /*****\
00121 *
00122 \*****/
00123

```

```

00124 #if defined(__clang__) || (defined(__GNUC__) && __GNUC__ > 6)
00125 /* suppress clang warning "shifting a negative signed value is
00126    undefined" in the assertions below */
00127 #pragma GCC diagnostic push
00128 #pragma GCC diagnostic ignored "-Wshift-negative-value"
00129 #endif
00130
00131 JAS_ATTRIBUTE_CONST
00132 JAS_ATTRIBUTE_DISABLE_USAN
00133 inline static int jas_int_asr(int x, unsigned n)
00134 {
00135     // Ensure that the shift of a negative value appears to behave as a
00136     // signed arithmetic shift.
00137     assert(((x > 0) && 1) == 1);
00138     // The behavior is undefined when x is negative. */
00139     // We tacitly assume the behavior is equivalent to a signed
00140     // arithmetic right shift.
00141     return x >> n;
00142 }
00143
00144 JAS_ATTRIBUTE_CONST
00145 JAS_ATTRIBUTE_DISABLE_USAN
00146 inline static int jas_int_asl(int x, unsigned n)
00147 {
00148     // Ensure that the shift of a negative value appears to behave as a
00149     // signed arithmetic shift.
00150     assert(((x < 0) && 1) == -1);
00151     // The behavior is undefined when x is negative. */
00152     // We tacitly assume the behavior is equivalent to a signed
00153     // arithmetic left shift.
00154     return x << n;
00155 }
00156
00157 JAS_ATTRIBUTE_CONST
00158 JAS_ATTRIBUTE_DISABLE_USAN
00159 inline static int_least32_t jas_least32_asr(int_least32_t x, unsigned n)
00160 {
00161     // Ensure that the shift of a negative value appears to behave as a
00162     // signed arithmetic shift.
00163     assert(((JAS_CAST(int_least32_t, x) > 0) && 1) == 1);
00164     // The behavior is undefined when x is negative. */
00165     // We tacitly assume the behavior is equivalent to a signed
00166     // arithmetic right shift.
00167     return x >> n;
00168 }
00169
00170 JAS_ATTRIBUTE_CONST
00171 JAS_ATTRIBUTE_DISABLE_USAN
00172 inline static int_least32_t jas_least32_asl(int_least32_t x, unsigned n)
00173 {
00174     // Ensure that the shift of a negative value appears to behave as a
00175     // signed arithmetic shift.
00176     assert(((JAS_CAST(int_least32_t, x) < 0) && 1) == -1);
00177     // The behavior is undefined when x is negative. */
00178     // We tacitly assume the behavior is equivalent to a signed
00179     // arithmetic left shift.
00180     return x << n;
00181 }
00182
00183 JAS_ATTRIBUTE_CONST
00184 JAS_ATTRIBUTE_DISABLE_USAN
00185 inline static int_fast32_t jas_fast32_asr(int_fast32_t x, unsigned n)
00186 {
00187     // Ensure that the shift of a negative value appears to behave as a
00188     // signed arithmetic shift.
00189     assert(((JAS_CAST(int_fast32_t, x) > 0) && 1) == 1);
00190     // The behavior is undefined when x is negative. */
00191     // We tacitly assume the behavior is equivalent to a signed
00192     // arithmetic right shift.
00193     return x >> n;
00194 }
00195
00196 JAS_ATTRIBUTE_CONST
00197 JAS_ATTRIBUTE_DISABLE_USAN
00198 inline static int_fast32_t jas_fast32_asl(int_fast32_t x, unsigned n)
00199 {
00200     // Ensure that the shift of a negative value appears to behave as a
00201     // signed arithmetic shift.
00202     assert(((JAS_CAST(int_fast32_t, x) < 0) && 1) == -1);
00203     // The behavior is undefined when x is negative. */
00204     // We tacitly assume the behavior is equivalent to a signed

```

```

00205         // arithmetic left shift.
00206         return x « n;
00207     }
00208
00209     #if defined(__clang__) || (defined(__GNUC__) && __GNUC__ > 6)
00210     #pragma GCC diagnostic pop
00211     #endif
00212
00213     /*****
00214     * Safe integer arithmetic (i.e., with overflow checking).
00215     *****/
00216
00217     /* Compute the product of two size_t integers with overflow checking. */
00218     inline static bool jas_safe_size_mul(size_t x, size_t y, size_t *result)
00219     {
00220         #if jas_has_builtin(__builtin_mul_overflow) || (defined(__GNUC__) && __GNUC__ > 5)
00221             size_t result_buffer;
00222             if (!result)
00223                 result = &result_buffer;
00224             return !__builtin_mul_overflow(x, y, result);
00225         #else
00226             /* Check if overflow would occur */
00227             if (x && y > SIZE_MAX / x) {
00228                 /* Overflow would occur. */
00229                 return false;
00230             }
00231             if (result) {
00232                 *result = x * y;
00233             }
00234             return true;
00235         #endif
00236     }
00237
00238     /* Compute the product of three size_t integers with overflow checking. */
00239     inline static bool jas_safe_size_mul3(size_t a, size_t b, size_t c,
00240         size_t *result)
00241     {
00242         size_t tmp;
00243         if (!jas_safe_size_mul(a, b, &tmp) ||
00244             !jas_safe_size_mul(tmp, c, &tmp)) {
00245             return false;
00246         }
00247         if (result) {
00248             *result = tmp;
00249         }
00250         return true;
00251     }
00252
00253     /* Compute the sum of two size_t integers with overflow checking. */
00254     inline static bool jas_safe_size_add(size_t x, size_t y, size_t *result)
00255     {
00256         #if jas_has_builtin(__builtin_add_overflow) || (defined(__GNUC__) && __GNUC__ > 5)
00257             size_t result_buffer;
00258             if (!result)
00259                 result = &result_buffer;
00260             return !__builtin_add_overflow(x, y, result);
00261         #else
00262             if (y > SIZE_MAX - x) {
00263                 return false;
00264             }
00265             if (result) {
00266                 *result = x + y;
00267             }
00268             return true;
00269         #endif
00270     }
00271
00272     /* Compute the difference of two size_t integers with overflow checking. */
00273     inline static bool jas_safe_size_sub(size_t x, size_t y, size_t *result)
00274     {
00275         #if jas_has_builtin(__builtin_sub_overflow) || (defined(__GNUC__) && __GNUC__ > 5)
00276             size_t result_buffer;
00277             if (!result)
00278                 result = &result_buffer;
00279             return !__builtin_sub_overflow(x, y, result);
00280         #else
00281             if (y > x) {
00282                 return false;
00283             }
00284             if (result) {
00285                 *result = x - y;

```

```

00286     }
00287     return true;
00288 #endif
00289 }
00290
00291 /* Compute the product of two int_fast32_t integers with overflow checking. */
00292 inline static bool jas_safe_intfast32_mul(int_fast32_t x, int_fast32_t y,
00293     int_fast32_t *result)
00294 {
00295     #if jas_has_builtin(__builtin_mul_overflow) || (defined(__GNUC__) && __GNUC__ > 5)
00296         int_fast32_t result_buffer;
00297         if (!result)
00298             result = &result_buffer;
00299         return !__builtin_mul_overflow(x, y, result);
00300     #else
00301         if (x > 0) {
00302             /* x is positive */
00303             if (y > 0) {
00304                 /* x and y are positive */
00305                 if (x > INT_FAST32_MAX / y) {
00306                     return false;
00307                 }
00308             } else {
00309                 /* x positive, y nonpositive */
00310                 if (y < INT_FAST32_MIN / x) {
00311                     return false;
00312                 }
00313             }
00314         } else {
00315             /* x is nonpositive */
00316             if (y > 0) {
00317                 /* x is nonpositive, y is positive */
00318                 if (x < INT_FAST32_MIN / y) {
00319                     return false;
00320                 }
00321             } else {
00322                 /* x and y are nonpositive */
00323                 if (x != 0 && y < INT_FAST32_MAX / x) {
00324                     return false;
00325                 }
00326             }
00327         }
00328         if (result) {
00329             *result = x * y;
00330         }
00331         return true;
00332     #endif
00333 }
00334
00335 /* Compute the product of three int_fast32_t integers with overflow checking. */
00336 inline static bool jas_safe_intfast32_mul3(int_fast32_t a, int_fast32_t b,
00337     int_fast32_t c, int_fast32_t *result)
00338 {
00339     int_fast32_t tmp;
00340     if (!jas_safe_intfast32_mul(a, b, &tmp) ||
00341         !jas_safe_intfast32_mul(tmp, c, &tmp)) {
00342         return false;
00343     }
00344     if (result) {
00345         *result = tmp;
00346     }
00347     return true;
00348 }
00349
00350 /* Compute the sum of two int_fast32_t integers with overflow checking. */
00351 inline static bool jas_safe_intfast32_add(int_fast32_t x, int_fast32_t y,
00352     int_fast32_t *result)
00353 {
00354     #if jas_has_builtin(__builtin_add_overflow) || (defined(__GNUC__) && __GNUC__ > 5)
00355         int_fast32_t result_buffer;
00356         if (!result)
00357             result = &result_buffer;
00358         return !__builtin_add_overflow(x, y, result);
00359     #else
00360         if ((y > 0 && x > INT_FAST32_MAX - y) ||
00361             (y < 0 && x < INT_FAST32_MIN - y)) {
00362             return false;
00363         }
00364         if (result) {
00365             *result = x + y;
00366         }

```

```

00367         return true;
00368 #endif
00369 }
00370
00371 #if 0
00372 /*
00373 This function is potentially useful but not currently used.
00374 So, it is commented out.
00375 */
00376 inline static bool jas_safe_uint_mul(unsigned x, unsigned y, unsigned *result)
00377 {
00378     /* Check if overflow would occur */
00379     if (x && y > UINT_MAX / x) {
00380         /* Overflow would occur. */
00381         return false;
00382     }
00383     if (result) {
00384         *result = x * y;
00385     }
00386     return true;
00387 }
00388 #endif
00389
00390 #ifdef __cplusplus
00391 }
00392 #endif
00393
00394 #endif

```

## 11.25 jas\_seq.h File Reference

Sequence/Matrix Library.

```

#include <jasper/jas_config.h>
#include <jasper/jas_types.h>
#include <jasper/jas_math.h>
#include <stdio.h>

```

### 11.25.1 Detailed Description

Sequence/Matrix Library.

## 11.26 jas\_seq.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *

```

```

00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_SEQ_H
00070 #define JAS_SEQ_H
00071
00072 /*****
00073  * Includes.
00074  *****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h> /* IWYU pragma: keep */
00078
00079 #include <jasper/jas_types.h>
00080 #include <jasper/jas_math.h>
00081
00082 #include <stdio.h>
00083
00084 #ifdef __cplusplus
00085 extern "C" {
00086 #endif
00087
00088 /*****
00089  * Constants.
00090  *****/
00091
00092 /* This matrix is a reference to another matrix. */
00093 #define JAS_MATRIX_REF 0x0001
00094
00095 /*****
00096  * Types.
00097  *****/
00098
00099 /* An element in a sequence. */
00100 #ifdef JAS_ENABLE_32BIT
00101 typedef int_least32_t jas_seqent_t;

```



```

00102 #define PRIjas_seqent PRIiLEAST32
00103 #else
00104 typedef int_fast32_t jas_seqent_t;
00105 #define PRIjas_seqent PRIiFAST32
00106 #endif
00107
00108 /* An element in a matrix. */
00109 #ifdef JAS_ENABLE_32BIT
00110 typedef int_least32_t jas_matent_t;
00111 #else
00112 typedef int_fast32_t jas_matent_t;
00113 #endif
00114
00115 #ifdef JAS_ENABLE_32BIT
00116 typedef int_least32_t jas_matind_t;
00117 #else
00118 typedef int_fast32_t jas_matind_t;
00119 #endif
00120
00121 /* Matrix. */
00122
00123 typedef struct {
00124
00125     /* Additional state information. */
00126     int flags_;
00127
00128     /* The starting horizontal index. */
00129     jas_matind_t xstart_;
00130
00131     /* The starting vertical index. */
00132     jas_matind_t ystart_;
00133
00134     /* The ending horizontal index. */
00135     jas_matind_t xend_;
00136
00137     /* The ending vertical index. */
00138     jas_matind_t yend_;
00139
00140     /* The number of rows in the matrix. */
00141     jas_matind_t numRows_;
00142
00143     /* The number of columns in the matrix. */
00144     jas_matind_t numcols_;
00145
00146     /* Pointers to the start of each row. */
00147     jas_seqent_t **rows_;
00148
00149     /* The allocated size of the rows array. */
00150     int_fast32_t maxrows_;
00151
00152     /* The matrix data buffer. */
00153     jas_seqent_t *data_;
00154
00155     /* The allocated size of the data array. */
00156     int_fast32_t datasize_;
00157 } jas_matrix_t;
00158
00159
00160 typedef jas_matrix_t jas_seq2d_t;
00161 typedef jas_matrix_t jas_seq_t;
00162
00163 /*****
00164  * Functions/macros for matrix class.
00165  *****/
00166
00167 /* Get the number of rows. */
00168 JAS_ATTRIBUTE_PURE
00169 static inline jas_matind_t jas_matrix_numrows(const jas_matrix_t *matrix)
00170 {
00171     return matrix->numrows_;
00172 }
00173
00174 /* Get the number of columns. */
00175 JAS_ATTRIBUTE_PURE
00176 static inline jas_matind_t jas_matrix_numcols(const jas_matrix_t *matrix)
00177 {
00178     return matrix->numcols_;
00179 }
00180
00181 JAS_ATTRIBUTE_PURE
00182 static inline jas_matind_t jas_matrix_size(const jas_matrix_t *matrix)

```

```

00183 {
00184     return jas_matrix_numcols(matrix) * jas_matrix_numrows(matrix);
00185 }
00186
00187 JAS_ATTRIBUTE_PURE
00188 static inline bool jas_matrix_empty(const jas_matrix_t *matrix)
00189 {
00190     return jas_matrix_numcols(matrix) == 0 || jas_matrix_numrows(matrix) == 0;
00191 }
00192
00193 /* Get a matrix element. */
00194 JAS_ATTRIBUTE_PURE
00195 static inline jas_segent_t jas_matrix_get(const jas_matrix_t *matrix, jas_matind_t i, jas_matind_t j)
00196 {
00197     return matrix->rows_[i][j];
00198 }
00199
00200 /* Set a matrix element. */
00201 static inline void jas_matrix_set(jas_matrix_t *matrix, jas_matind_t i, jas_matind_t j, jas_segent_t v)
00202 {
00203     matrix->rows_[i][j] = v;
00204 }
00205
00206 /* Get an element from a matrix that is known to be a row or column vector. */
00207 JAS_ATTRIBUTE_PURE
00208 static inline jas_segent_t jas_matrix_getv(const jas_matrix_t *matrix, jas_matind_t i)
00209 {
00210     return matrix->numrows_ == 1
00211         ? matrix->rows_[0][i]
00212         : matrix->rows_[i][0];
00213 }
00214
00215 /* Set an element in a matrix that is known to be a row or column vector. */
00216 static inline void jas_matrix_setv(jas_matrix_t *matrix, jas_matind_t i, jas_segent_t v)
00217 {
00218     if (matrix->numrows_ == 1)
00219         matrix->rows_[0][i] = v;
00220     else
00221         matrix->rows_[i][0] = v;
00222 }
00223
00224 /* Get the address of an element in a matrix. */
00225 JAS_ATTRIBUTE_PURE
00226 static inline jas_segent_t *jas_matrix_getref(const jas_matrix_t *matrix, jas_matind_t i, jas_matind_t j)
00227 {
00228     return &matrix->rows_[i][j];
00229 }
00230
00231 JAS_ATTRIBUTE_PURE
00232 static inline jas_segent_t *jas_matrix_getvref(const jas_matrix_t *matrix, jas_matind_t i)
00233 {
00234     return matrix->numrows_ > 1
00235         ? jas_matrix_getref(matrix, i, 0)
00236         : jas_matrix_getref(matrix, 0, i);
00237 }
00238
00239 /* Create a matrix with the specified dimensions. */
00240 JAS_DLLEXPORT jas_matrix_t *jas_matrix_create(jas_matind_t numrows, jas_matind_t numcols);
00241
00242 /* Destroy a matrix. */
00243 JAS_DLLEXPORT void jas_matrix_destroy(jas_matrix_t *matrix);
00244
00245 /* Resize a matrix. The previous contents of the matrix are lost. */
00246 JAS_DLLEXPORT int jas_matrix_resize(jas_matrix_t *matrix, jas_matind_t numrows, jas_matind_t numcols);
00247
00248 JAS_DLLEXPORT int jas_matrix_output(jas_matrix_t *matrix, FILE *out);
00249
00250 /* Create a matrix that references part of another matrix. */
00251 JAS_DLLEXPORT int jas_matrix_bindsub(jas_matrix_t *mat0, jas_matrix_t *mat1, jas_matind_t r0,
00252     jas_matind_t c0, jas_matind_t r1, jas_matind_t c1);
00253
00254 /* Create a matrix that is a reference to a row of another matrix. */
00255 static inline int jas_matrix_bindrow(jas_matrix_t *mat0, jas_matrix_t *mat1, jas_matind_t r)
00256 {
00257     return jas_matrix_bindsub(mat0, mat1, r, 0, r, mat1->numcols_ - 1);
00258 }
00259
00260 /* Create a matrix that is a reference to a column of another matrix. */
00261 static inline int jas_matrix_bindcol(jas_matrix_t *mat0, jas_matrix_t *mat1, jas_matind_t c)
00262 {
00263     return jas_matrix_bindsub(mat0, mat1, 0, c, mat1->numrows_ - 1, c);

```

```

00264 }
00265
00266 /* Clip the values of matrix elements to the specified range. */
00267 JAS_DLLEXPORT void jas_matrix_clip(jas_matrix_t *matrix, jas_sequent_t minval,
00268     jas_sequent_t maxval);
00269
00270 /* Arithmetic shift left of all elements in a matrix. */
00271 JAS_DLLEXPORT void jas_matrix_asl(jas_matrix_t *matrix, unsigned n);
00272
00273 /* Arithmetic shift right of all elements in a matrix. */
00274 JAS_DLLEXPORT void jas_matrix_asr(jas_matrix_t *matrix, unsigned n);
00275
00276 /* Almost-but-not-quite arithmetic shift right of all elements in a matrix. */
00277 JAS_DLLEXPORT void jas_matrix_divpow2(jas_matrix_t *matrix, unsigned n);
00278
00279 /* Set all elements of a matrix to the specified value. */
00280 JAS_DLLEXPORT void jas_matrix_setall(jas_matrix_t *matrix, jas_sequent_t val);
00281
00282 /* The spacing between rows of a matrix. */
00283 JAS_ATTRIBUTE_PURE
00284 static inline size_t jas_matrix_rowstep(const jas_matrix_t *matrix)
00285 {
00286     return matrix->numrows_ > 1
00287         ? (size_t)(matrix->rows_[1] - matrix->rows_[0])
00288         : 0u;
00289 }
00290
00291 /* The spacing between columns of a matrix. */
00292 JAS_ATTRIBUTE_PURE
00293 static inline size_t jas_matrix_step(const jas_matrix_t *matrix)
00294 {
00295     return matrix->numrows_ > 1
00296         ? jas_matrix_rowstep(matrix)
00297         : 1;
00298 }
00299
00300 /* Compare two matrices for equality. */
00301 JAS_DLLEXPORT int jas_matrix_cmp(jas_matrix_t *mat0, jas_matrix_t *mat1);
00302
00303 JAS_DLLEXPORT jas_matrix_t *jas_matrix_copy(jas_matrix_t *x);
00304
00305 JAS_DLLEXPORT jas_matrix_t *jas_matrix_input(FILE *);
00306
00307 JAS_ATTRIBUTE_CONST
00308 static inline jas_sequent_t jas_sequent_asl(jas_sequent_t x, unsigned n)
00309 {
00310     #ifdef JAS_ENABLE_32BIT
00311         return jas_least32_asl(x, n);
00312     #else
00313         return jas_fast32_asl(x, n);
00314     #endif
00315 }
00316
00317 JAS_ATTRIBUTE_CONST
00318 static inline jas_sequent_t jas_sequent_asr(jas_sequent_t x, unsigned n)
00319 {
00320     #ifdef JAS_ENABLE_32BIT
00321         return jas_least32_asr(x, n);
00322     #else
00323         return jas_fast32_asr(x, n);
00324     #endif
00325 }
00326
00327 /*****
00328  * Functions/macros for 2-D sequence class.
00329  *****/
00330
00331 JAS_DLLEXPORT jas_seq2d_t *jas_seq2d_copy(jas_seq2d_t *x);
00332
00333 JAS_DLLEXPORT jas_matrix_t *jas_seq2d_create(jas_matind_t xstart, jas_matind_t ystart,
00334     jas_matind_t xend, jas_matind_t yend);
00335
00336 static inline void jas_seq2d_destroy(jas_seq2d_t *s)
00337 {
00338     jas_matrix_destroy(s);
00339 }
00340
00341 JAS_ATTRIBUTE_PURE
00342 static inline jas_matind_t jas_seq2d_xstart(const jas_seq2d_t *s)
00343 {
00344     return s->xstart_;

```

```

00345 }
00346
00347 JAS_ATTRIBUTE_PURE
00348 static inline jas_matind_t jas_seq2d_ystart(const jas_seq2d_t *s)
00349 {
00350     return s->ystart_;
00351 }
00352
00353 JAS_ATTRIBUTE_PURE
00354 static inline jas_matind_t jas_seq2d_xend(const jas_seq2d_t *s)
00355 {
00356     return s->xend_;
00357 }
00358
00359 JAS_ATTRIBUTE_PURE
00360 static inline jas_matind_t jas_seq2d_yend(const jas_seq2d_t *s)
00361 {
00362     return s->yend_;
00363 }
00364
00365 JAS_ATTRIBUTE_PURE
00366 static inline jas_seqent_t *jas_seq2d_getref(const jas_seq2d_t *s, jas_matind_t x, jas_matind_t y)
00367 {
00368     return jas_matrix_getref(s, y - s->ystart_, x - s->xstart_);
00369 }
00370
00371 JAS_ATTRIBUTE_PURE
00372 static inline jas_seqent_t jas_seq2d_get(const jas_seq2d_t *s, jas_matind_t x, jas_matind_t y)
00373 {
00374     return jas_matrix_get(s, y - s->ystart_, x - s->xstart_);
00375 }
00376
00377 JAS_ATTRIBUTE_PURE
00378 static inline size_t jas_seq2d_rowstep(const jas_seq2d_t *s)
00379 {
00380     return jas_matrix_rowstep(s);
00381 }
00382
00383 JAS_ATTRIBUTE_PURE
00384 static inline unsigned jas_seq2d_width(const jas_seq2d_t *s)
00385 {
00386     return (unsigned)(s->xend_ - s->xstart_);
00387 }
00388
00389 JAS_ATTRIBUTE_PURE
00390 static inline unsigned jas_seq2d_height(const jas_seq2d_t *s)
00391 {
00392     return (unsigned)(s->yend_ - s->ystart_);
00393 }
00394
00395 static inline void jas_seq2d_setshift(jas_seq2d_t *s, jas_matind_t x, jas_matind_t y)
00396 {
00397     s->xstart_ = x;
00398     s->ystart_ = y;
00399     s->xend_ = s->xstart_ + s->numcols_;
00400     s->yend_ = s->ystart_ + s->numrows_;
00401 }
00402
00403 JAS_ATTRIBUTE_PURE
00404 static inline jas_matind_t jas_seq2d_size(const jas_seq2d_t *s)
00405 {
00406     return jas_seq2d_width(s) * jas_seq2d_height(s);
00407 }
00408
00409 JAS_ATTRIBUTE_PURE
00410 static inline bool jas_seq2d_empty(const jas_seq2d_t *s)
00411 {
00412     return jas_seq2d_width(s) == 0 || jas_seq2d_height(s) == 0;
00413 }
00414
00415 JAS_DLLEXPORT int jas_seq2d_bindsub(jas_matrix_t *s, jas_matrix_t *sl, jas_matind_t xstart,
00416     jas_matind_t ystart, jas_matind_t xend, jas_matind_t yend);
00417
00418 /*****
00419 * Functions/macros for 1-D sequence class.
00420 *****/
00421
00422 static inline jas_seq_t *jas_seq_create(jas_matind_t start, jas_matind_t end)
00423 {
00424     return jas_seq2d_create(start, 0, end, 1);
00425 }

```

```

00426
00427 static inline void jas_seq_destroy(jas_seq_t *seq)
00428 {
00429     jas_seq2d_destroy(seq);
00430 }
00431
00432 static inline void jas_seq_set(jas_seq_t *seq, jas_matind_t i, jas_sequent_t v)
00433 {
00434     seq->rows_[0][i - seq->xstart_] = v;
00435 }
00436
00437 JAS_ATTRIBUTE_PURE
00438 static inline jas_sequent_t *jas_seq_getref(const jas_seq_t *seq, jas_matind_t i)
00439 {
00440     return &seq->rows_[0][i - seq->xstart_];
00441 }
00442
00443 JAS_ATTRIBUTE_PURE
00444 static inline jas_sequent_t jas_seq_get(const jas_seq_t *seq, jas_matind_t i)
00445 {
00446     return seq->rows_[0][i - seq->xstart_];
00447 }
00448
00449 JAS_ATTRIBUTE_PURE
00450 static inline jas_matind_t jas_seq_start(const jas_seq_t *seq)
00451 {
00452     return seq->xstart_;
00453 }
00454
00455 JAS_ATTRIBUTE_PURE
00456 static inline jas_matind_t jas_seq_end(const jas_seq_t *seq)
00457 {
00458     return seq->xend_;
00459 }
00460
00461 #ifdef __cplusplus
00462 }
00463 #endif
00464
00465 #endif

```

## 11.27 jas\_stream.h File Reference

I/O Stream Class.

```

#include <jasper/jas_config.h>
#include <stdio.h>
#include <jasper/jas_types.h>

```

### Macros

- #define [jas\\_stream\\_eof](#)(stream) (((stream)->flags\_ & JAS\_STREAM\_EOF) != 0)  
*Get the EOF indicator for a stream.*
- #define [jas\\_stream\\_error](#)(stream) (((stream)->flags\_ & JAS\_STREAM\_ERR) != 0)  
*Get the error indicator for a stream.*
- #define [jas\\_stream\\_clearerr](#)(stream) ((stream)->flags\_ &= ~(JAS\_STREAM\_ERR | JAS\_STREAM\_EOF))  
*Clear the error indicator for a stream.*
- #define [jas\\_stream\\_getrwlimit](#)(stream) (((const jas\_stream\_t \*) (stream))->rwlimit\_)  
*Get the read/write limit for a stream.*
- #define [jas\\_stream\\_getrwcoun](#)(stream) (((const jas\_stream\_t \*) (stream))->rwcnt\_)  
*Get the read/write count for a stream.*

- #define [jas\\_stream\\_getc](#)(stream) [jas\\_stream\\_getc\\_func](#)(stream)  
*jas\_stream\_getc Read a character from a stream.*
- #define [jas\\_stream\\_putc](#)(stream, c) [jas\\_stream\\_putc\\_func](#)(stream, c)  
*jas\_stream\_putc Write a character to a stream.*
- #define [jas\\_stream\\_peekc](#)(stream)  
*Look at the next character to be read from a stream without actually removing the character from the stream.*

## Functions

- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_fopen](#) (const char \*filename, const char \*mode)  
*Open a file as a stream.*
- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_memopen](#) (char \*buffer, int buffer\_size)  
*Open a memory buffer as a stream.*
- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_memopen2](#) (char \*buffer, size\_t buffer\_size)
- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_fdopen](#) (int fd, const char \*mode)  
*Open a file descriptor as a stream.*
- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_freopen](#) (const char \*path, const char \*mode, FILE \*fp)  
*Open a stdio (i.e., C standard library) stream as a stream.*
- JAS\_DLLEXPORT [jas\\_stream\\_t](#) \* [jas\\_stream\\_tmpfile](#) (void)  
*Open a temporary file as a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_close](#) ([jas\\_stream\\_t](#) \*stream)  
*Close a stream.*
- JAS\_DLLEXPORT long [jas\\_stream\\_setrwlimit](#) ([jas\\_stream\\_t](#) \*stream, long rwlimit)  
*Set the read/write limit for a stream.*
- JAS\_DLLEXPORT long [jas\\_stream\\_setrwcoun](#)t ([jas\\_stream\\_t](#) \*stream, long rw\_count)  
*Set the read/write count for a stream.*
- JAS\_DLLEXPORT unsigned [jas\\_stream\\_read](#) ([jas\\_stream\\_t](#) \*stream, void \*buffer, unsigned count)  
*Read characters from a stream into a buffer.*
- JAS\_DLLEXPORT unsigned [jas\\_stream\\_peek](#) ([jas\\_stream\\_t](#) \*stream, void \*buffer, size\_t count)  
*Attempt to retrieve one or more pending characters of input from a stream into a buffer without actually removing the characters from the stream.*
- JAS\_DLLEXPORT unsigned [jas\\_stream\\_write](#) ([jas\\_stream\\_t](#) \*stream, const void \*buffer, unsigned count)  
*Write characters from a buffer to a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_printf](#) ([jas\\_stream\\_t](#) \*stream, const char \*format,...)  
*Write formatted output to a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_puts](#) ([jas\\_stream\\_t](#) \*stream, const char \*s)  
*Write a string to a stream.*
- JAS\_DLLEXPORT char \* [jas\\_stream\\_gets](#) ([jas\\_stream\\_t](#) \*stream, char \*buffer, int buffer\_size)  
*Read a line of input from a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_ungetc](#) ([jas\\_stream\\_t](#) \*stream, int c)  
*Put a character back on a stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT int [jas\\_stream\\_isseekable](#) ([jas\\_stream\\_t](#) \*stream)  
*Determine if stream supports seeking.*
- JAS\_DLLEXPORT long [jas\\_stream\\_seek](#) ([jas\\_stream\\_t](#) \*stream, long offset, int origin)  
*Set the current position within the stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT long [jas\\_stream\\_tell](#) ([jas\\_stream\\_t](#) \*stream)  
*Get the current position within the stream.*

- JAS\_DLLEXPORT int [jas\\_stream\\_rewind](#) (jas\_stream\_t \*stream)  
*Seek to the beginning of a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_flush](#) (jas\_stream\_t \*stream)  
*Flush any pending output to a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_copy](#) (jas\_stream\_t \*destination, jas\_stream\_t \*source, int count)  
*Copy data from one stream to another.*
- JAS\_DLLEXPORT int [jas\\_stream\\_display](#) (jas\_stream\_t \*stream, FILE \*fp, int count)  
*Print a hex dump of data read from a stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_gobble](#) (jas\_stream\_t \*stream, int count)  
*Consume (i.e., discard) characters from stream.*
- JAS\_DLLEXPORT int [jas\\_stream\\_pad](#) (jas\_stream\_t \*stream, int count, int value)  
*Write a fill character multiple times to a stream.*
- JAS\_ATTRIBUTE\_PURE JAS\_DLLEXPORT long [jas\\_stream\\_length](#) (jas\_stream\_t \*stream)  
*Get the size of the file associated with the specified stream.*

### 11.27.1 Detailed Description

I/O Stream Class.

## 11.28 jas\_stream.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A

```

```

00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.  IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.  NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE.  AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY.  THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES").  THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_STREAM_H
00070 #define JAS_STREAM_H
00071
00072 /*****
00073  * Includes.
00074  *****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h> /* IWYU pragma: export */
00078
00079 #include <stdio.h>
00080 #if defined(JAS_HAVE_FCNTL_H)
00081 #include <fcntl.h>
00082 #endif
00083 #include <jasper/jas_types.h>
00084
00085 #ifdef __cplusplus
00086 extern "C" {
00087 #endif
00088
00094 /*****
00095  * Constants.
00096  *****/
00097
00098 /* On most UNIX systems, we probably need to define O_BINARY ourselves. */
00099 #ifndef O_BINARY
00100 #define O_BINARY      0
00101 #endif
00102
00103 /*
00104  * Stream open flags.
00105  */
00106
00107 /* The stream was opened for reading. */
00108 #define JAS_STREAM_READ 0x0001
00109 /* The stream was opened for writing. */
00110 #define JAS_STREAM_WRITE 0x0002
00111 /* The stream was opened for appending. */
00112 #define JAS_STREAM_APPEND 0x0004
00113 /* The stream was opened in binary mode. */
00114 #define JAS_STREAM_BINARY 0x0008
00115 /* The stream should be created/truncated. */
00116 #define JAS_STREAM_CREATE 0x0010
00117
00118 /*
00119  * Stream buffering flags.
00120  */
00121
00122 /* The stream is unbuffered. */
00123 #define JAS_STREAM_UNBUF 0x0000
00124 /* The stream is line buffered. */
00125 #define JAS_STREAM_LINEBUF 0x0001
00126 /* The stream is fully buffered. */
00127 #define JAS_STREAM_FULLBUF 0x0002
00128 /* The buffering mode mask. */
00129 #define JAS_STREAM_BUFMODEMASK 0x000f

```



```

00130
00131 /* The memory associated with the buffer needs to be deallocated when the
00132    stream is destroyed. */
00133 #define JAS_STREAM_FREEBUF      0x0008
00134 /* The buffer is currently being used for reading. */
00135 #define JAS_STREAM_RDBUF       0x0010
00136 /* The buffer is currently being used for writing. */
00137 #define JAS_STREAM_WRBUF       0x0020
00138
00139 /*
00140  * Stream error flags.
00141  */
00142
00143 /* The end-of-file has been encountered (on reading). */
00144 #define JAS_STREAM_EOF  0x0001
00145 /* An I/O error has been encountered on the stream. */
00146 #define JAS_STREAM_ERR  0x0002
00147 /* The read/write limit has been exceeded. */
00148 #define JAS_STREAM_RWLIMIT  0x0004
00149 /* The error mask. */
00150 #define JAS_STREAM_ERRMASK \
00151     (JAS_STREAM_EOF | JAS_STREAM_ERR | JAS_STREAM_RWLIMIT)
00152
00153 /*
00154  * Other miscellaneous constants.
00155  */
00156
00157 /* The default buffer size (for fully-buffered operation). */
00158 #define JAS_STREAM_BUFSIZE  8192
00159 /* The default permission mask for file creation. */
00160 #define JAS_STREAM_PERMS    0666
00161
00162 /* The maximum number of characters that can always be put back on a stream. */
00163 #define JAS_STREAM_MAXPUTBACK  16
00164
00165 /*****\
00166  * Types.
00167  \*****/
00168
00169 /*
00170  * Generic file object.
00171  */
00172
00173 typedef void jas_stream_obj_t;
00174
00175 /*
00176  * Generic file object operations.
00177  */
00178
00179 typedef struct {
00180
00181     /* Read characters from a file object. */
00182     int (*read_)(jas_stream_obj_t *obj, char *buf, unsigned cnt);
00183
00184     /* Write characters to a file object. */
00185     int (*write_)(jas_stream_obj_t *obj, const char *buf, unsigned cnt);
00186
00187     /* Set the position for a file object. */
00188     long (*seek_)(jas_stream_obj_t *obj, long offset, int origin);
00189
00190     /* Close a file object. */
00191     int (*close_)(jas_stream_obj_t *obj);
00192
00193 } jas_stream_ops_t;
00194
00195 /*
00196  * Stream object.
00197  */
00198
00199 typedef struct {
00200
00201     /* The mode in which the stream was opened. */
00202     int openmode_;
00203
00204     /* The buffering mode. */
00205     int bufmode_;
00206
00207     /* The stream status. */
00208     int flags_;
00209
00210     /* The start of the buffer area to use for reading/writing. */

```

```

00211     jas_uchar *bufbase_;
00212
00213     /* The start of the buffer area excluding the extra initial space for
00214        character putback. */
00215     jas_uchar *bufstart_;
00216
00217     /* The buffer size. */
00218     int bufsize_;
00219
00220     /* The current position in the buffer. */
00221     jas_uchar *ptr_;
00222
00223     /* The number of characters that must be read/written before
00224        the buffer needs to be filled/flushed. */
00225     int cnt_;
00226
00227     /* A trivial buffer to be used for unbuffered operation. */
00228     jas_uchar tinybuf_[JAS_STREAM_MAXPUTBACK + 1];
00229
00230     /* The operations for the underlying stream file object. */
00231     const jas_stream_ops_t *ops_;
00232
00233     /* The underlying stream file object. */
00234     jas_stream_obj_t *obj_;
00235
00236     /* The number of characters read/written. */
00237     long rwcnt_;
00238
00239     /* The maximum number of characters that may be read/written. */
00240     long rwlimit_;
00241
00242 } jas_stream_t;
00243
00244 /*
00245  * Regular file object.
00246  */
00247
00248 /*
00249  * File descriptor file object.
00250  */
00251 typedef struct {
00252     int fd;
00253     int flags;
00254     char pathname[L_tmpnam + 1];
00255 } jas_stream_fileobj_t;
00256
00257 /* Delete underlying file object upon stream close. */
00258 #define JAS_STREAM_FILEOBJ_DELCLOSE 0x01
00259 /* Do not close underlying file object upon stream close. */
00260 #define JAS_STREAM_FILEOBJ_NOCLOSE 0x02
00261
00262 /*
00263  * Memory file object.
00264  */
00265
00266 typedef struct {
00267
00268     /* The data associated with this file. */
00269     jas_uchar *buf_;
00270
00271     /* The allocated size of the buffer for holding file data. */
00272     size_t bufsize_;
00273
00274     /* The length of the file. */
00275     uint_fast32_t len_;
00276
00277     /* The seek position. */
00278     uint_fast32_t pos_;
00279
00280     /* Is the buffer growable? */
00281     int growable_;
00282
00283     /* Was the buffer allocated internally? */
00284     int myalloc_;
00285
00286 } jas_stream_memobj_t;
00287
00288 /*****\
00289  * Macros/functions for opening and closing streams.
00290  \*****/
00291

```

```

00306 JAS_DLLEXPORT
00307 jas_stream_t *jas_stream_fopen(const char *filename, const char *mode);
00308
00343 JAS_DLLEXPORT
00344 jas_stream_t *jas_stream_memopen(char *buffer, int buffer_size);
00345
00351 JAS_DLLEXPORT
00352 jas_stream_t *jas_stream_memopen2(char *buffer, size_t buffer_size);
00353
00368 JAS_DLLEXPORT
00369 jas_stream_t *jas_stream_fdopen(int fd, const char *mode);
00370
00393 JAS_DLLEXPORT
00394 jas_stream_t *jas_stream_freopen(const char *path, const char *mode, FILE *fp);
00395
00410 JAS_DLLEXPORT jas_stream_t *jas_stream_tmpfile(void);
00411
00430 JAS_DLLEXPORT
00431 int jas_stream_close(jas_stream_t *stream);
00432
00433 /*****
00434  * Macros/functions for getting/setting the stream state.
00435  */
00436
00447 #define jas_stream_eof(stream) \
00448     (((stream)->flags_ & JAS_STREAM_EOF) != 0)
00449
00460 #define jas_stream_error(stream) \
00461     (((stream)->flags_ & JAS_STREAM_ERR) != 0)
00462
00471 #define jas_stream_clearerr(stream) \
00472     (((stream)->flags_ & ~(JAS_STREAM_ERR | JAS_STREAM_EOF))
00473
00484 #define jas_stream_getrwlmit(stream) \
00485     (((const jas_stream_t *) (stream))->rwlmit_)
00486
00501 JAS_DLLEXPORT long jas_stream_setrwlmit(jas_stream_t *stream, long rwlmit);
00502
00513 #define jas_stream_getrwcunt(stream) \
00514     (((const jas_stream_t *) (stream))->rwcnt_)
00515
00528 JAS_DLLEXPORT long jas_stream_setrwcunt(jas_stream_t *stream, long rw_cunt);
00529
00530 /*****
00531  * Macros/functions for I/O.
00532  */
00533
00534 /* Read a character from a stream. */
00535 #ifndef NDEBUG
00540 #define jas_stream_getc(stream) jas_stream_getc_func(stream)
00541 #else
00542 #define jas_stream_getc(stream) jas_stream_getc_macro(stream)
00543 #endif
00544
00545 /* Write a character to a stream. */
00546 #ifndef NDEBUG
00551 #define jas_stream_putc(stream, c) jas_stream_putc_func(stream, c)
00552 #else
00553 #define jas_stream_putc(stream, c) jas_stream_putc_macro(stream, c)
00554 #endif
00555
00593 JAS_DLLEXPORT
00594 unsigned jas_stream_read(jas_stream_t *stream, void *buffer, unsigned count);
00595
00619 JAS_DLLEXPORT
00620 unsigned jas_stream_peek(jas_stream_t *stream, void *buffer, size_t count);
00621
00653 JAS_DLLEXPORT
00654 unsigned jas_stream_write(jas_stream_t *stream, const void *buffer,
00655     unsigned count);
00656
00678 JAS_DLLEXPORT
00679 int jas_stream_printf(jas_stream_t *stream, const char *format, ...);
00680
00696 JAS_DLLEXPORT int jas_stream_puts(jas_stream_t *stream, const char *s);
00697
00719 JAS_DLLEXPORT char *
00720 jas_stream_gets(jas_stream_t *stream, char *buffer, int buffer_size);
00721
00740 #define jas_stream_peekc(stream) \
00741     (((stream)->cnt_ <= 0) ? jas_stream_fillbuf(stream, 0) : \

```

```

00742         ((int) (*(stream)->ptr_)))
00743
00767 JAS_DLLEXPORT int
00768 jas_stream_ungetc(jas_stream_t *stream, int c);
00769
00770 /*****
00771 * Macros/functions for getting/setting the stream position.
00772 \*****/
00773
00789 JAS_ATTRIBUTE_PURE JAS_DLLEXPORT
00790 int jas_stream_isseekable(jas_stream_t *stream);
00791
00811 JAS_DLLEXPORT
00812 long jas_stream_seek(jas_stream_t *stream, long offset, int origin);
00813
00826 JAS_ATTRIBUTE_PURE JAS_DLLEXPORT
00827 long jas_stream_tell(jas_stream_t *stream);
00828
00841 JAS_DLLEXPORT int jas_stream_rewind(jas_stream_t *stream);
00842
00843 /*****
00844 * Macros/functions for flushing.
00845 \*****/
00846
00857 JAS_DLLEXPORT int jas_stream_flush(jas_stream_t *stream);
00858
00859 /*****
00860 * Miscellaneous macros/functions.
00861 \*****/
00862
00883 JAS_DLLEXPORT int jas_stream_copy(jas_stream_t *destination, jas_stream_t *source, int count);
00884
00907 JAS_DLLEXPORT
00908 int jas_stream_display(jas_stream_t *stream, FILE *fp, int count);
00909
00932 JAS_DLLEXPORT int jas_stream_gobble(jas_stream_t *stream, int count);
00933
00958 JAS_DLLEXPORT int jas_stream_pad(jas_stream_t *stream, int count, int value);
00959
00977 JAS_ATTRIBUTE_PURE JAS_DLLEXPORT
00978 long jas_stream_length(jas_stream_t *stream);
00979
00980 /*****
00981 * Internal functions.
00982 \*****/
00983
00984 /* The following functions are for internal use only! If you call them
00985 directly, you will die a horrible, miserable, and painful death! */
00986
00987 /* These prototypes need to be here for the sake of the stream_getc and
00988 stream_putc macros. */
00989 /* Library users must not invoke these functions directly. */
00990 JAS_DLLEXPORT int jas_stream_fillbuf(jas_stream_t *stream, int getflag);
00991 JAS_DLLEXPORT int jas_stream_flushbuf(jas_stream_t *stream, int c);
00992 JAS_DLLEXPORT int jas_stream_getc_func(jas_stream_t *stream);
00993 JAS_DLLEXPORT int jas_stream_putc_func(jas_stream_t *stream, int c);
00994
00995 /* Read a character from a stream. */
00996 static inline int jas_stream_getc2(jas_stream_t *stream)
00997 {
00998     if (--stream->cnt_ < 0)
00999         return jas_stream_fillbuf(stream, 1);
01000
01001     ++stream->rwcnt_;
01002     return (int) (*(stream->ptr_++));
01003 }
01004
01005 static inline int jas_stream_getc_macro(jas_stream_t *stream)
01006 {
01007     if (stream->flags_ & (JAS_STREAM_ERR | JAS_STREAM_EOF | JAS_STREAM_RWLIMIT))
01008         return EOF;
01009
01010     if (stream->rwlimit_ >= 0 && stream->rwcnt_ >= stream->rwlimit_) {
01011         stream->flags_ |= JAS_STREAM_RWLIMIT;
01012         return EOF;
01013     }
01014
01015     return jas_stream_getc2(stream);
01016 }
01017
01018 /* Write a character to a stream. */

```

```

01019 static inline int jas_stream_putc2(jas_stream_t *stream, jas_uchar c)
01020 {
01021     stream->bufmode_ |= JAS_STREAM_WRBUF;
01022
01023     if (--stream->cnt_ < 0)
01024         return jas_stream_flushbuf(stream, c);
01025     else {
01026         ++stream->rwcnt_;
01027         return (int)(*stream->ptr_++ = c);
01028     }
01029 }
01030
01031 static inline int jas_stream_putc_macro(jas_stream_t *stream, jas_uchar c)
01032 {
01033     if (stream->flags_ & (JAS_STREAM_ERR | JAS_STREAM_EOF | JAS_STREAM_RWLIMIT))
01034         return EOF;
01035
01036     if (stream->rwlimit_ >= 0 && stream->rwcnt_ >= stream->rwlimit_) {
01037         stream->flags_ |= JAS_STREAM_RWLIMIT;
01038         return EOF;
01039     }
01040
01041     return jas_stream_putc2(stream, c);
01042 }
01043
01048 #ifdef __cplusplus
01049 }
01050 #endif
01051
01052 #endif

```

## 11.29 jas\_string.h File Reference

String Library.

```
#include <jasper/jas_config.h>
```

### 11.29.1 Detailed Description

String Library.

## 11.30 jas\_string.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the

```

```

00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_STRING_H
00070 #define JAS_STRING_H
00071
00072 /*****
00073  * Includes.
00074  *****/
00075
00076 /* The configuration header file should be included first. */
00077 #include <jasper/jas_config.h>
00078
00079 #ifdef __cplusplus
00080 extern "C" {
00081 #endif
00082
00083 /*****
00084  * Functions.
00085  *****/
00086
00087 /* Copy a string (a la strdup). */
00088 JAS_DLLEXPORT char *jas_strdup(const char *);
00089
00090 #ifdef __cplusplus
00091 }
00092 #endif
00093
00094 #endif

```

## 11.31 jas\_tmr.h File Reference

Timer Code.

```
#include <jasper/jas_config.h>
#include <time.h>
```

### 11.31.1 Detailed Description

Timer Code.

## 11.32 jas\_tmr.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 2004 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
```

```
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00067 #ifndef JAS_TMR_H
00068 #define JAS_TMR_H
00069
00070 /* The configuration header file should be included first. */
00071 #include <jasper/jas_config.h>
00072
00073 #if defined(JAS_HAVE_SYS_TIME_H)
00074 #include <sys/time.h>
00075 #else
00076 #include <time.h>
00077 #endif
00078
00079 #ifdef __cplusplus
00080 extern "C" {
00081 #endif
00082
00083 #if defined(JAS_HAVE_GETTIMEOFDAY)
00084
00085 typedef struct {
00086     struct timeval start;
00087     struct timeval stop;
00088 } jas_tmr_t;
00089
00090 #elif defined(JAS_HAVE_GETRUSAGE)
00091
00092 typedef struct {
00093     struct rusage start;
00094     struct rusage stop;
00095 } jas_tmr_t;
00096
00097 #else
00098
00099 typedef int jas_tmr_t;
00100
00101 #endif
00102
00103 JAS_DLLEXPORT void jas_tmr_start(jas_tmr_t *tmr);
00104 JAS_DLLEXPORT void jas_tmr_stop(jas_tmr_t *tmr);
00105 JAS_DLLEXPORT double jas_tmr_get(jas_tmr_t *tmr);
00106
00107 #ifdef __cplusplus
00108 }
00109 #endif
00110
00111 #endif
```

## 11.33 jas\_tvp.h File Reference

Tag/Value Pair Parser.

```
#include <jasper/jas_config.h>
```

### 11.33.1 Detailed Description

Tag/Value Pair Parser.



## 11.34 jas\_tvp.h

[Go to the documentation of this file.](#)

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef JAS_TVP_H
00063 #define JAS_TVP_H
00064
00065 /*****
00066  * Includes.
00067  *****/
00068 #include <jasper/jas_config.h>
00069
00070 #ifdef __cplusplus
00071 extern "C" {
00072 #endif
00073
00074 /*****

```

```

00082 * Types.
00083 \*****
00084
00085 /* Tag information type. */
00086
00087 typedef struct {
00088     int id;
00089     /* The ID for the tag. */
00090
00091     const char *name;
00092     /* The name of the tag. */
00093 } jas_taginfo_t;
00094
00095 /* Tag-value parser type. */
00096
00097 typedef struct {
00098     char *buf;
00099     /* The parsing buffer. */
00100
00101     char *tag;
00102     /* The current tag name. */
00103
00104     const char *val;
00105     /* The current value. */
00106
00107     char *pos;
00108     /* The current position in the parsing buffer. */
00109 } jas_tvparser_t;
00110
00111 \*****
00112 * Tag information functions.
00113 \*****
00114
00115 /* Lookup a tag by name. */
00116 JAS_ATTRIBUTE_PURE
00117 JAS_DLLEXPORT const jas_taginfo_t *jas_taginfos_lookup(const jas_taginfo_t *taginfos, const char *name);
00118
00119 /* This function returns a pointer to the specified taginfo object if it
00120    exists (i.e., the pointer is nonnull); otherwise, a pointer to a dummy
00121    object is returned. This is useful in some situations to avoid checking
00122    for a null pointer. */
00123 JAS_ATTRIBUTE_PURE
00124 JAS_DLLEXPORT const jas_taginfo_t *jas_taginfo_nonnull(const jas_taginfo_t *taginfo);
00125
00126 \*****
00127 * Tag-value parser functions.
00128 \*****
00129
00130 /* Create a tag-value parser for the specified string. */
00131 JAS_DLLEXPORT jas_tvparser_t *jas_tvparser_create(const char *s);
00132
00133 /* Destroy a tag-value parser. */
00134 JAS_DLLEXPORT void jas_tvparser_destroy(jas_tvparser_t *tvparser);
00135
00136 /* Get the next tag-value pair. */
00137 JAS_DLLEXPORT int jas_tvparser_next(jas_tvparser_t *tvparser);
00138
00139 /* Get the tag name for the current tag-value pair. */
00140 JAS_ATTRIBUTE_PURE
00141 JAS_DLLEXPORT const char *jas_tvparser_gettag(const jas_tvparser_t *tvparser);
00142
00143 /* Get the value for the current tag-value pair. */
00144 JAS_ATTRIBUTE_PURE
00145 JAS_DLLEXPORT const char *jas_tvparser_getval(const jas_tvparser_t *tvparser);
00146
00147 #ifdef __cplusplus
00148 }
00149 #endif
00150
00151 #endif

```

## 11.35 jas\_types.h File Reference

Primitive Types.

```
#include <jasper/jas_config.h>
#include <stddef.h>
#include <stdint.h>
#include <stdbool.h>
#include <inttypes.h>
```

### 11.35.1 Detailed Description

Primitive Types.

## 11.36 jas\_types.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
```

```

00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_TYPES_H
00070 #define JAS_TYPES_H
00071
00072 /* The configuration header file should be included first. */
00073 #include <jasper/jas_config.h>
00074
00075 /* Note: The immediately following header files should eventually be removed. */
00076 #include <stddef.h> /* IWYU pragma: export */
00077 #include <stdint.h> /* IWYU pragma: export */
00078
00079 #if defined(JAS_HAVE_SYS_TYPES_H)
00080 #include <sys/types.h> /* IWYU pragma: export */
00081 #endif
00082
00083 #define jas_uchar unsigned char
00084 #define jas_uint unsigned int
00085 #define jas_ulong unsigned long
00086 #define jas_longlong long long
00087 #define jas_ulonglong unsigned long long
00088
00089 #if defined(_MSC_VER) && (_MSC_VER < 1800)
00090 #define bool int
00091 #define false 0
00092 #define true 1
00093
00094 #define PRIxFAST32 "x"
00095 #define PRIxFAST16 PRIxFAST32
00096 #define PRIuFAST32 "u"
00097 #define PRIuFAST16 PRIuFAST32
00098 #define PRIiFAST32 "i"
00099 #ifdef _WIN64
00100     #define PRIdPTR "lld"
00101 #else
00102     #define PRIdPTR "d"
00103 #endif
00104
00105 #ifndef _HUGE_ENUF
00106     #define _HUGE_ENUF 1e+300
00107 #endif
00108
00109 #define INFINITY ((float)( _HUGE_ENUF * _HUGE_ENUF))
00110
00111 #define strtoull _strtoui64
00112
00113 #else
00114 #include <stdbool.h> /* IWYU pragma: export */
00115 #include <inttypes.h> /* IWYU pragma: export */
00116 #endif
00117
00118 /* The below macro is intended to be used for type casts. By using this
00119  * macro, type casts can be easily located in the source code with
00120  * tools like "grep". */
00121 #define JAS_CAST(t, e) \
00122     ((t) (e))
00123
00124 /* The number of bits in the integral type uint_fast32_t. */
00125 /* NOTE: This could underestimate the size on some exotic architectures. */
00126 #define JAS_UINTFAST32_NUMBITS (8 * sizeof(uint_fast32_t))
00127
00128 #ifdef __cplusplus
00129 extern "C" {
00130 #endif
00131
00132 #ifdef __cplusplus

```

```
00133 }
00134 #endif
00135
00136 #endif
```

## 11.37 jas\_version.h File Reference

JasPer Version.

```
#include <jasper/jas_config.h>
```

### 11.37.1 Detailed Description

JasPer Version.

## 11.38 jas\_version.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
```

```

00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00069 #ifndef JAS_VERSION_H
00070 #define JAS_VERSION_H
00071
00072 /* The configuration header file should be included first. */
00073 #include <jasper/jas_config.h>
00074
00075 #ifdef __cplusplus
00076 extern "C" {
00077 #endif
00078
00079 /*****
00080  * Constants and types.
00081  \*****/
00082
00083 #if !defined(JAS_VERSION)
00084 /* The version information below should match that specified in
00085  the "configure.in" file! */
00086 #define JAS_VERSION "unknown"
00087 #endif
00088
00089 #define JAS_COPYRIGHT \
00090     "Copyright (c) 2001-2006 Michael David Adams.\n" \
00091     "Copyright (c) 1999-2000 Image Power, Inc. and the University of\n" \
00092     "  British Columbia.\n" \
00093     "All rights reserved.\n"
00094
00095 #define JAS_NOTES \
00096     "For more information about this software, please visit the following\n" \
00097     "web sites/pages:\n" \
00098     "  http://www.ece.uvic.ca/~mdadams/jasper\n" \
00099     "  http://www.jpeg.org/software\n" \
00100     "To be added to the (moderated) JasPer software announcements\n" \
00101     "mailing list, send an email to:\n" \
00102     "  jasper-announce-subscribe@yahoogroups.com\n" \
00103     "To be added to the (unmoderated) JasPer software discussion\n" \
00104     "mailing list, send an email to:\n" \
00105     "  jasper-discussion-subscribe@yahoogroups.com\n" \
00106     "Please send any bug reports to:\n" \
00107     "  mdadams@ieee.org\n"
00108
00109 /*****
00110  * Functions.
00111  \*****/
00112
00113 JAS_ATTRIBUTE_CONST
00114 JAS_DLLEXPORT const char *jas_getversion(void);
00115 /* Get the version information for the JasPer library. */
00116 /* Note: Since libjasper can be built as a shared library, the version
00117  returned by this function may not necessarily correspond to JAS_VERSION. */
00118
00119 #ifdef __cplusplus
00120 }
00121 #endif
00122
00123 #endif

```

## 11.39 jasper.h File Reference

JasPer Main Header.

```
#include <jasper/jas_config.h>
#include <jasper/jas_types.h>
#include <jasper/jas_version.h>
#include <jasper/jas_init.h>
#include <jasper/jas_cm.h>
#include <jasper/jas_icc.h>
#include <jasper/jas_fix.h>
#include <jasper/jas_debug.h>
#include <jasper/jas_getopt.h>
#include <jasper/jas_image.h>
#include <jasper/jas_math.h>
#include <jasper/jas_malloc.h>
#include <jasper/jas_seq.h>
#include <jasper/jas_stream.h>
#include <jasper/jas_string.h>
#include <jasper/jas_tmr.h>
#include <jasper/jas_tvp.h>
```

### 11.39.1 Detailed Description

JasPer Main Header.

## 11.40 jasper.h

[Go to the documentation of this file.](#)

```
00001 /*
00002  * Copyright (c) 2001-2003 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
```

```

00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00067 #ifndef JAS_JASPER_H
00068 #define JAS_JASPER_H
00069
00070 // IWYU pragma: begin_exports
00071
00072 /* The configuration header file should be included first. */
00073 #include <jasper/jas_config.h>
00074
00075 #include <jasper/jas_types.h>
00076 #include <jasper/jas_version.h>
00077
00078 #include <jasper/jas_init.h>
00079 #include <jasper/jas_cm.h>
00080 #include <jasper/jas_icc.h>
00081 #include <jasper/jas_fix.h>
00082 #include <jasper/jas_debug.h>
00083 #include <jasper/jas_getopt.h>
00084 #include <jasper/jas_image.h>
00085 #include <jasper/jas_math.h>
00086 #include <jasper/jas_malloc.h>
00087 #include <jasper/jas_seq.h>
00088 #include <jasper/jas_stream.h>
00089 #include <jasper/jas_string.h>
00090 #include <jasper/jas_tmr.h>
00091 #include <jasper/jas_tvp.h>
00092
00093 #ifdef __cplusplus
00094 extern "C" {
00095 #endif
00096
00097 #ifdef __cplusplus
00098 }
00099 #endif
00100
00101 // IWYU pragma: end_exports
00102
00103 #endif

```

## 11.41 jp2\_cod.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  * British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.

```



```

00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * JP2 Library
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JP2_COD_H
00071 #define JP2_COD_H
00072
00073 /*****\
00074  * Includes.
00075  \*****/
00076
00077 #include "jasper/jas_types.h"
00078 #include "jasper/jas_stream.h"
00079 #include "jasper/jas_image.h"
00080
00081 #include <stdio.h>
00082
00083 /*****\
00084  * Macros.
00085  \*****/

```

```

00086
00087 #define JP2_SPTOBPC(s, p) \
00088     (((p) - 1) & 0x7f) | (((s) & 1) << 7))
00089
00090 /*****
00091  * Box class.
00092  *****/
00093
00094 #define JP2_BOX_HDRLEN(ext) ((ext) ? 16 : 8)
00095
00096 /* Box types. */
00097 #define JP2_BOX_JP          0x6a502020 /* Signature */
00098 #define JP2_BOX_FTYPE      0x66747970 /* File Type */
00099 #define JP2_BOX_JP2H       0x6a703268 /* JP2 Header */
00100 #define JP2_BOX_IHDR       0x69686472 /* Image Header */
00101 #define JP2_BOX_BPCC       0x62706363 /* Bits Per Component */
00102 #define JP2_BOX_COLR       0x636f6c72 /* Color Specification */
00103 #define JP2_BOX_PCLR       0x70636c72 /* Palette */
00104 #define JP2_BOX_CMAP       0x636d6170 /* Component Mapping */
00105 #define JP2_BOX_CDEF       0x63646566 /* Channel Definition */
00106 #define JP2_BOX_RES        0x72657320 /* Resolution */
00107 #define JP2_BOX_RESC       0x72657363 /* Capture Resolution */
00108 #define JP2_BOX_RESD       0x72657364 /* Default Display Resolution */
00109 #define JP2_BOX_JP2C       0x6a703263 /* Contiguous Code Stream */
00110 #define JP2_BOX_JP2I       0x6a703269 /* Intellectual Property */
00111 #define JP2_BOX_XML        0x786d6c20 /* XML */
00112 #define JP2_BOX_UUID       0x75756964 /* UUID */
00113 #define JP2_BOX_UINF       0x75696e66 /* UUID Info */
00114 #define JP2_BOX_ULST       0x75637374 /* UUID List */
00115 #define JP2_BOX_URL        0x75726c20 /* URL */
00116
00117 #define JP2_BOX_SUPER      0x01
00118 #define JP2_BOX_NODATA     0x02
00119
00120 /* JP box data. */
00121
00122 #define JP2_JP_MAGIC       0x0d0a870a
00123 #define JP2_JP_LEN        12
00124
00125 typedef struct {
00126     uint_fast32_t magic;
00127 } jp2_jp_t;
00128
00129 /* FTYPE box data. */
00130
00131 #define JP2_FTYPE_MAXCOMPATCODES 32
00132 #define JP2_FTYPE_MAJVER        0x6a703220
00133 #define JP2_FTYPE_MINVER        0
00134 #define JP2_FTYPE_COMPATCODE     JP2_FTYPE_MAJVER
00135
00136 typedef struct {
00137     uint_fast32_t majver;
00138     uint_fast32_t minver;
00139     uint_fast32_t numcompatcodes;
00140     uint_fast32_t compatcodes[JP2_FTYPE_MAXCOMPATCODES];
00141 } jp2_ftyp_t;
00142
00143 /* IHDR box data. */
00144
00145 #define JP2_IHDR_COMPTYPE      7
00146 #define JP2_IHDR_BPCNULL      255
00147
00148 typedef struct {
00149     uint_fast32_t width;
00150     uint_fast32_t height;
00151     uint_fast16_t numcmpts;
00152     uint_fast8_t bpc;
00153     uint_fast8_t comptype;
00154     uint_fast8_t csunk;
00155     uint_fast8_t ipr;
00156 } jp2_ihdr_t;
00157
00158 /* BPCC box data. */
00159
00160 typedef struct {
00161     uint_fast16_t numcmpts;
00162     uint_fast8_t *bpcs;
00163 } jp2_bpcc_t;
00164
00165 /* COLR box data. */
00166

```

```

00167 #define JP2_COLR_ENUM    1
00168 #define JP2_COLR_ICC      2
00169 #define JP2_COLR_PRI      0
00170
00171 #define JP2_COLR_SRGB     16
00172 #define JP2_COLR_SGRAY    17
00173 #define JP2_COLR_SYCC     18
00174
00175 typedef struct {
00176     uint_fast8_t method;
00177     uint_fast8_t pri;
00178     uint_fast8_t approx;
00179     uint_fast32_t csid;
00180     uint_fast8_t *iccp;
00181     size_t iccplen;
00182     /* XXX - Someday we ought to add ICC profile data here. */
00183 } jp2_colr_t;
00184
00185 /* PCLR box data. */
00186
00187 typedef struct {
00188     uint_fast16_t numlutents;
00189     uint_fast8_t numchans;
00190     int_fast32_t *lutdata;
00191     uint_fast8_t *bpc;
00192 } jp2_pclr_t;
00193
00194 /* CDEF box per-channel data. */
00195
00196 #define JP2_CDEF_RGB_R    1
00197 #define JP2_CDEF_RGB_G    2
00198 #define JP2_CDEF_RGB_B    3
00199
00200 #define JP2_CDEF_YCBCR_Y    1
00201 #define JP2_CDEF_YCBCR_CB    2
00202 #define JP2_CDEF_YCBCR_CR    3
00203
00204 #define JP2_CDEF_GRAY_Y    1
00205
00206 #define JP2_CDEF_TYPE_COLOR    0
00207 #define JP2_CDEF_TYPE_OPACITY    1
00208 #define JP2_CDEF_TYPE_UNSPEC    65535
00209 #define JP2_CDEF_ASOC_ALL    0
00210 #define JP2_CDEF_ASOC_NONE    65535
00211
00212 typedef struct {
00213     uint_fast16_t channo;
00214     uint_fast16_t type;
00215     uint_fast16_t assoc;
00216 } jp2_cdefchan_t;
00217
00218 /* CDEF box data. */
00219
00220 typedef struct {
00221     uint_fast16_t numchans;
00222     jp2_cdefchan_t *ents;
00223 } jp2_cdef_t;
00224
00225 typedef struct {
00226     uint_fast16_t cmptno;
00227     uint_fast8_t map;
00228     uint_fast8_t pcol;
00229 } jp2_cmapent_t;
00230
00231 typedef struct {
00232     uint_fast16_t numchans;
00233     jp2_cmapent_t *ents;
00234 } jp2_cmap_t;
00235
00236 #define JP2_CMAP_DIRECT    0
00237 #define JP2_CMAP_PALETTE    1
00238
00239 /* Generic box. */
00240
00241 struct jp2_boxops_s;
00242 typedef struct {
00243     const struct jp2_boxops_s *ops;
00244     const struct jp2_boxinfo_s *info;
00245     uint_fast32_t type;

```

```

00248
00249     /* The length of the box including the (variable-length) header. */
00250     uint_fast32_t len;
00251
00252     /* The length of the box data. */
00253     uint_fast32_t datalen;
00254
00255     union {
00256         jp2_jp_t jp;
00257         jp2_ftyp_t ftyp;
00258         jp2_ihdr_t ihdr;
00259         jp2_bpcc_t bpcc;
00260         jp2_colr_t colr;
00261         jp2_pclr_t pclr;
00262         jp2_cdef_t cdef;
00263         jp2_cmap_t cmap;
00264     } data;
00265
00266 } jp2_box_t;
00267
00268 typedef struct jp2_boxops_s {
00269     void (*init)(jp2_box_t *box);
00270     void (*destroy)(jp2_box_t *box);
00271     int (*getdata)(jp2_box_t *box, jas_stream_t *in);
00272     int (*putdata)(const jp2_box_t *box, jas_stream_t *out);
00273     void (*dumpdata)(const jp2_box_t *box, FILE *out);
00274 } jp2_boxops_t;
00275
00276 /*****
00277  *
00278  \*****/
00279
00280 typedef struct jp2_boxinfo_s {
00281     int type;
00282     int flags;
00283     const char *name;
00284     jp2_boxops_t ops;
00285 } jp2_boxinfo_t;
00286
00287 /*****
00288  * Box class.
00289  \*****/
00290
00291 jp2_box_t *jp2_box_create(int type);
00292 void jp2_box_destroy(jp2_box_t *box);
00293 jp2_box_t *jp2_box_get(jas_stream_t *in);
00294 int jp2_box_put(jp2_box_t *box, jas_stream_t *out);
00295
00296 JAS_ATTRIBUTE_CONST
00297 static inline uint_least8_t JP2_DTYPETOBPC(uint_least8_t dtype)
00298 {
00299     return (JAS_IMAGE_CDT_GETSGND(dtype) << 7) | (JAS_IMAGE_CDT_GETPREC(dtype) - 1);
00300 }
00301
00302 JAS_ATTRIBUTE_CONST
00303 static inline uint_least8_t JP2_BPCTODTYPE(uint_least8_t bpc)
00304 {
00305     return JAS_IMAGE_CDT_SETSGND(bpc >> 7) | JAS_IMAGE_CDT_SETPREC((bpc & 0x7f) + 1);
00306 }
00307
00308 #define ICC_CS_RGB      0x52474220
00309 #define ICC_CS_YCBCR    0x59436272
00310 #define ICC_CS_GRAY     0x47524159
00311
00312 const jp2_cdefchan_t *jp2_cdef_lookup(jp2_cdef_t *cdef, int channo);
00313
00314
00315 #endif

```

## 11.42 jp2\_dec.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */

```

```

00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * Jasper License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 #ifndef JP2_DEC_H
00065 #define JP2_DEC_H
00066
00067 #include "jasper/jas_image.h"
00068 #include "jp2_cod.h"
00069
00070 typedef struct {
00071
00072     jp2_box_t *pclr;
00073     jp2_box_t *cdef;
00074     jp2_box_t *ihdr;
00075     jp2_box_t *bpcc;
00076     jp2_box_t *cmap;
00077     jp2_box_t *colr;
00078     jas_image_t *image;
00079     uint_fast16_t numchans;
00080     uint_fast16_t *chantocmptlut;
00081
00082 } jp2_dec_t;
00083
00084 #endif

```

## 11.43 jpc\_bs.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Bit Stream Class
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_BS_H
00071 #define JPC_BS_H
00072
00073 /*****\
00074  * Includes.
00075  */
00076 /*****\
00077 #include "jasper/jas_types.h"
00078 #include "jasper/jas_stream.h"

```

```

00079
00080 #include <assert.h>
00081 #include <stdio.h>
00082
00083 /*****
00084  * Constants.
00085  *****/
00086
00087 /*
00088  * Bit stream open mode flags.
00089  */
00090
00091 /* Bit stream open for reading. */
00092 #define JPC_BITSTREAM_READ    0x01
00093 /* Bit stream open for writing. */
00094 #define JPC_BITSTREAM_WRITE   0x02
00095
00096 /*
00097  * Bit stream flags.
00098  */
00099
00100 /* Do not close underlying character stream. */
00101 #define JPC_BITSTREAM_NOCLOSE 0x01
00102 /* End of file has been reached while reading. */
00103 #define JPC_BITSTREAM_EOF     0x02
00104 /* An I/O error has occurred. */
00105 #define JPC_BITSTREAM_ERR     0x04
00106
00107 /*****
00108  * Types.
00109  *****/
00110
00111 /* Bit stream class. */
00112
00113 typedef struct {
00114
00115     /* Some miscellaneous flags. */
00116     int flags_;
00117
00118     /* The input/output buffer. */
00119     uint_fast16_t buf_;
00120
00121     /* The number of bits remaining in the byte being read/written. */
00122     int cnt_;
00123
00124     /* The underlying stream associated with this bit stream. */
00125     jas_stream_t *stream_;
00126
00127     /* The mode in which this bit stream was opened. */
00128     int openmode_;
00129 } jpc_bitstream_t;
00130
00131
00132 /*****
00133  * Functions/macros for opening and closing bit streams..
00134  *****/
00135
00136 /* Open a stream as a bit stream. */
00137 jpc_bitstream_t *jpc_bitstream_sopen(jas_stream_t *stream, const char *mode);
00138
00139 /* Close a bit stream. */
00140 int jpc_bitstream_close(jpc_bitstream_t *bitstream);
00141
00142 /*****
00143  * Functions/macros for reading from and writing to bit streams..
00144  *****/
00145
00146 /* Read a bit from a bit stream. */
00147 #ifndef NDEBBUG
00148 #define jpc_bitstream_getbit(bitstream) \
00149     jpc_bitstream_getbit_func(bitstream)
00150 #else
00151 #define jpc_bitstream_getbit(bitstream) \
00152     jpc_bitstream_getbit_macro(bitstream)
00153 #endif
00154
00155 /* Write a bit to a bit stream. */
00156 #ifndef NDEBBUG
00157 #define jpc_bitstream_putbit(bitstream, v) \
00158     jpc_bitstream_putbit_func(bitstream, v)
00159 #else

```

```

00160 #define jpc_bitstream_putbit(bitstream, v) \
00161     jpc_bitstream_putbit_macro(bitstream, v)
00162 #endif
00163
00164 /* Read one or more bits from a bit stream. */
00165 long jpc_bitstream_getbits(jpc_bitstream_t *bitstream, int n);
00166
00167 /* Write one or more bits to a bit stream. */
00168 int jpc_bitstream_putbits(jpc_bitstream_t *bitstream, int n, long v);
00169
00170 /*****
00171  * Functions/macros for flushing and aligning bit streams.
00172  *****/
00173
00174 /* Align the current position within the bit stream to the next byte
00175    boundary. */
00176 int jpc_bitstream_align(jpc_bitstream_t *bitstream);
00177
00178 /* Align the current position in the bit stream with the next byte boundary,
00179    ensuring that certain bits consumed in the process match a particular
00180    pattern. */
00181 int jpc_bitstream_inalign(jpc_bitstream_t *bitstream, int fillmask,
00182    int filldata);
00183
00184 /* Align the current position in the bit stream with the next byte boundary,
00185    writing bits from the specified pattern (if necessary) in the process. */
00186 int jpc_bitstream_outalign(jpc_bitstream_t *bitstream, int filldata);
00187
00188 /* Check if a bit stream needs alignment. */
00189 JAS_ATTRIBUTE_PURE
00190 int jpc_bitstream_needalign(const jpc_bitstream_t *bitstream);
00191
00192 /* How many additional bytes would be output if the bit stream was aligned? */
00193 JAS_ATTRIBUTE_PURE
00194 int jpc_bitstream_pending(const jpc_bitstream_t *bitstream);
00195
00196 /*****
00197  * Functions/macros for querying state information for bit streams.
00198  *****/
00199
00200 /* Has EOF been encountered on a bit stream? */
00201 #define jpc_bitstream_eof(bitstream) \
00202     ((bitstream)->flags_ & JPC_BITSTREAM_EOF)
00203
00204 /*****
00205  * Internals.
00206  *****/
00207
00208 /* DO NOT DIRECTLY INVOKE ANY OF THE MACROS OR FUNCTIONS BELOW. THEY ARE
00209    FOR INTERNAL USE ONLY. */
00210
00211 int jpc_bitstream_getbit_func(jpc_bitstream_t *bitstream);
00212
00213 int jpc_bitstream_putbit_func(jpc_bitstream_t *bitstream, int v);
00214
00215 int jpc_bitstream_fillbuf(jpc_bitstream_t *bitstream);
00216
00217 #define jpc_bitstream_getbit_macro(bitstream) \
00218     (assert((bitstream)->openmode_ & JPC_BITSTREAM_READ), \
00219      (--(bitstream)->cnt_ >= 0) ? \
00220      ((int)((bitstream)->buf_ > (bitstream)->cnt_) & 1) : \
00221      jpc_bitstream_fillbuf(bitstream))
00222
00223 #define jpc_bitstream_putbit_macro(bitstream, bit) \
00224     (assert((bitstream)->openmode_ & JPC_BITSTREAM_WRITE), \
00225      (--(bitstream)->cnt_ < 0) ? \
00226      ((bitstream)->buf_ = ((bitstream)->buf_ < 8) & 0xffff, \
00227      (bitstream)->cnt_ = ((bitstream)->buf_ == 0xffff) ? 6 : 7, \
00228      (bitstream)->buf_ |= ((bit) & 1) << (bitstream)->cnt_, \
00229      (jas_stream_putc((bitstream)->stream_, (bitstream)->buf_ > 8) == EOF) \
00230      ? (EOF) : ((bit) & 1)) : \
00231      ((bitstream)->buf_ |= ((bit) & 1) << (bitstream)->cnt_, \
00232      (bit) & 1))
00233
00234 #endif

```



## 11.44 jpc\_cod.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * $Id$
00066  */
00067
00068 #ifndef JPC_COD_H
00069 #define JPC_COD_H
00070
00071 /*****
00072  * Constants.
00073  *****/
00074
00075 /* The nominal word size used by this implementation. */
00076 #define JPC_PREC      32
00077
00078 #endif

```

## 11.45 jpc\_cs.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * JPEG-2000 Code Stream Library
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_CS_H
00071 #define JPC_CS_H
00072
00073 /*****\
00074  * Includes.
00075  */
00076
00077 #include "jasper/jas_stream.h"
00078

```

```

00079 #include <assert.h>
00080 #include <stdio.h>
00081
00082 /*****
00083  * Constants and Types.
00084  *****/
00085
00086 /* The maximum number of resolution levels. */
00087 #define JPC_MAXRLVLS    33
00088
00089 /* The maximum number of bands. */
00090 #define JPC_MAXBANDS    (3 * JPC_MAXRLVLS + 1)
00091
00092 /* The maximum number of layers. */
00093 #define JPC_MAXLYRS     16384
00094
00095 /*****
00096  * Code stream.
00097  *****/
00098
00099 /*
00100  * Code stream states.
00101  */
00102
00103 /* Initial. */
00104 #define JPC_CS_INIT      0
00105 /* Main header. */
00106 #define JPC_CS_MHDR      1
00107 /* Tile-part header. */
00108 #define JPC_CS_THDR      2
00109 /* Main trailer. */
00110 #define JPC_CS_MTLR      3
00111 /* Tile-part data. */
00112 #define JPC_CS_TDATA     4
00113
00114 /*
00115  * Unfortunately, the code stream syntax was not designed in such a way that
00116  * any given marker segment can be correctly decoded without additional state
00117  * derived from previously decoded marker segments.
00118  * For example, a RGN/COC/QCC marker segment cannot be decoded unless the
00119  * number of components is known.
00120  */
00121
00122 /*
00123  * Code stream state information.
00124  */
00125
00126 typedef struct {
00127     /* The number of components. */
00128     uint_fast16_t numcomps;
00129 } jpc_cstate_t;
00130
00131
00132 /*****
00133  * SOT marker segment parameters.
00134  *****/
00135
00136 typedef struct {
00137     /* The tile number. */
00138     uint_fast16_t tileno;
00139
00140     /* The combined length of the marker segment and its auxiliary data
00141        (i.e., packet data). */
00142     uint_fast32_t len;
00143
00144     /* The tile-part instance. */
00145     uint_fast8_t partno;
00146
00147     /* The number of tile-parts. */
00148     uint_fast8_t numparts;
00149 } jpc_sot_t;
00150
00151
00152 /*****
00153  * SIZ marker segment parameters.
00154  *****/
00155
00156 /* Per component information. */
00157
00158
00159

```

```

00160 typedef struct {
00161     /* The precision of the samples. */
00162     uint_fast8_t prec;
00163     /* The signedness of the samples. */
00164     uint_fast8_t sgnd;
00165     /* The horizontal separation of samples with respect to the reference
00166        grid. */
00167     uint_fast8_t hsamp;
00168     /* The vertical separation of samples with respect to the reference
00169        grid. */
00170     uint_fast8_t vsamp;
00171 } jpc_sizcomp_t;
00172 /* SIZ marker segment parameters. */
00173 typedef struct {
00174     /* The code stream capabilities. */
00175     uint_fast16_t caps;
00176     /* The width of the image in units of the reference grid. */
00177     uint_fast32_t width;
00178     /* The height of the image in units of the reference grid. */
00179     uint_fast32_t height;
00180     /* The horizontal offset from the origin of the reference grid to the
00181        left side of the image area. */
00182     uint_fast32_t xoff;
00183     /* The vertical offset from the origin of the reference grid to the
00184        top side of the image area. */
00185     uint_fast32_t yoff;
00186     /* The nominal width of a tile in units of the reference grid. */
00187     uint_fast32_t tilewidth;
00188     /* The nominal height of a tile in units of the reference grid. */
00189     uint_fast32_t tileheight;
00190     /* The horizontal offset from the origin of the reference grid to the
00191        left side of the first tile. */
00192     uint_fast32_t tilexoff;
00193     /* The vertical offset from the origin of the reference grid to the
00194        top side of the first tile. */
00195     uint_fast32_t tileyoff;
00196     /* The number of components. */
00197     uint_fast16_t numcomps;
00198     /* The per-component information. */
00199     jpc_sizcomp_t *comps;
00200 } jpc_siz_t;
00201
00202 /*****
00203  * COD marker segment parameters.
00204  *****/
00205 /*
00206  * Coding style constants.
00207  */
00208 /* Precincts may be used. */
00209 #define JPC_COX_PRT 0x01
00210 /* SOP marker segments may be used. */
00211 #define JPC_COD_SOP 0x02
00212 /* EPH marker segments may be used. */
00213 #define JPC_COD_EPH 0x04
00214
00215 /*
00216  * Progression order constants.
00217  */
00218
00219 /* Layer-resolution-component-precinct progressive

```

```

00241     (i.e., progressive by fidelity). */
00242 #define JPC_COD_LRCPPRG 0
00243 /* Resolution-layer-component-precinct progressive
00244     (i.e., progressive by resolution). */
00245 #define JPC_COD_RLCPPRG 1
00246 /* Resolution-precinct-component-layer progressive. */
00247 #define JPC_COD_RPCLPRG 2
00248 /* Precinct-component-resolution-layer progressive. */
00249 #define JPC_COD_PCRLPRG 3
00250 /* Component-position-resolution-layer progressive. */
00251 #define JPC_COD_CPRLPRG 4
00252
00253 /*
00254  * Code block style constants.
00255  */
00256
00257 #define JPC_COX_LAZY      0x01 /* Selective arithmetic coding bypass. */
00258 #define JPC_COX_RESET    0x02 /* Reset context probabilities. */
00259 #define JPC_COX_TERMALL  0x04 /* Terminate all coding passes. */
00260 #define JPC_COX_VSC      0x08 /* Vertical stripe causal context formation. */
00261 #define JPC_COX_PTERM    0x10 /* Predictable termination. */
00262 #define JPC_COX_SEGSYM   0x20 /* Use segmentation symbols. */
00263
00264 /* Transform constants. */
00265 #define JPC_COX_INS       0x00 /* Irreversible 9/7. */
00266 #define JPC_COX_RFT      0x01 /* Reversible 5/3. */
00267
00268 /* Multicomponent transform constants. */
00269 #define JPC_COD_NOMCT     0x00 /* No multicomponent transform. */
00270 #define JPC_COD_MCT      0x01 /* Multicomponent transform. */
00271
00272 /* Get the code block size value from the code block size exponent. */
00273 JAS_ATTRIBUTE_CONST
00274 static inline unsigned JPC_COX_CBLKSIZEEXPN(unsigned x)
00275 {
00276     return x - 2;
00277 }
00278
00279 /* Get the code block size exponent from the code block size value. */
00280 JAS_ATTRIBUTE_CONST
00281 static inline unsigned JPC_COX_GETCBLKSIZEEXPN(unsigned x)
00282 {
00283     return x + 2;
00284 }
00285
00286 /* Per resolution-level information. */
00287 typedef struct {
00288
00289     /* The packet partition width. */
00290     uint_fast8_t parwidthval;
00291
00292     /* The packet partition height. */
00293     uint_fast8_t parheightval;
00294
00295 } jpc_coxrlvl_t;
00296
00297 /* Per component information. */
00298 typedef struct {
00299
00300     /* The coding style. */
00301     uint_fast8_t csty;
00302
00303     /* The number of decomposition levels. */
00304     uint_fast8_t numdlvls;
00305
00306     /* The nominal code block width specifier. */
00307     uint_fast8_t cblkwidthval;
00308
00309     /* The nominal code block height specifier. */
00310     uint_fast8_t cblkheightval;
00311
00312     /* The style of coding passes. */
00313     uint_fast8_t cblksty;
00314
00315     /* The QMFB employed. */
00316     uint_fast8_t qmfbid;
00317
00318     /* The number of resolution levels. */
00319     int numrlvls;
00320
00321 }

```

```

00322
00323     /* The per-resolution-level information. */
00324     jpc_coxrlvl_t rlvls[JPC_MAXRLVLS];
00325
00326 } jpc_coxcp_t;
00327
00328 /* COD marker segment parameters. */
00329
00330 typedef struct {
00331
00332     /* The general coding style. */
00333     uint_fast8_t csty;
00334
00335     /* The progression order. */
00336     uint_fast8_t prg;
00337
00338     /* The number of layers. */
00339     uint_fast16_t numlyrs;
00340
00341     /* The multicomponent transform. */
00342     uint_fast8_t mctrans;
00343
00344     /* Component-related parameters. */
00345     jpc_coxcp_t compparms;
00346
00347 } jpc_cod_t;
00348
00349 /* COC marker segment parameters. */
00350
00351 typedef struct {
00352
00353     /* The component number. */
00354     uint_fast16_t compno;
00355
00356     /* Component-related parameters. */
00357     jpc_coxcp_t compparms;
00358
00359 } jpc_coc_t;
00360
00361 /******\
00362 * RGN marker segment parameters.
00363 \*****/
00364
00365 /* The maxshift ROI style. */
00366 #define JPC_RGN_MAXSHIFT      0x00
00367
00368 typedef struct {
00369
00370     /* The component to which the marker applies. */
00371     uint_fast16_t compno;
00372
00373     /* The ROI style. */
00374     uint_fast8_t roisty;
00375
00376     /* The ROI shift value. */
00377     uint_fast8_t roishift;
00378
00379 } jpc_rgn_t;
00380
00381 /******\
00382 * QCD/QCC marker segment parameters.
00383 \*****/
00384
00385 /*
00386 * Quantization style constants.
00387 */
00388
00389 #define JPC_QCX_NOQNT      0 /* No quantization. */
00390 #define JPC_QCX_SIQNT      1 /* Scalar quantization, implicit. */
00391 #define JPC_QCX_SEQNT      2 /* Scalar quantization, explicit. */
00392
00393 /*
00394 * Stepsize manipulation macros.
00395 */
00396
00397 JAS_ATTRIBUTE_CONST
00398 static inline unsigned JPC_QCX_GETEXPN(unsigned x)
00399 {
00400     return x » 11;
00401 }
00402

```

```

00403 JAS_ATTRIBUTE_CONST
00404 static inline unsigned JPC_QCX_GETMANT(unsigned x)
00405 {
00406     return x & 0x7ff;
00407 }
00408
00409 JAS_ATTRIBUTE_CONST
00410 static inline uint_fast16_t JPC_QCX_EXPONENT(unsigned x)
00411 {
00412     assert(!(x & (~0x1f)));
00413
00414     return (x & 0x1f) << 11;
00415 }
00416
00417 JAS_ATTRIBUTE_CONST
00418 static inline uint_fast16_t JPC_QCX_MANT(unsigned x)
00419 {
00420     assert(!(x & (~0x7ff)));
00421
00422     return x & 0x7ff;
00423 }
00424
00425 /* Per component information. */
00426
00427 typedef struct {
00428     /* The quantization style. */
00429     uint_fast8_t qntsty;
00430
00431     /* The number of step sizes. */
00432     int numstepsizes;
00433
00434     /* The step sizes. */
00435     uint_fast16_t *stepsizes;
00436
00437     /* The number of guard bits. */
00438     uint_fast8_t numguard;
00439 } jpc_qcxcp_t;
00440
00441 /* QCC marker segment parameters. */
00442
00443 typedef struct {
00444     /* The component associated with this marker segment. */
00445     uint_fast16_t compno;
00446
00447     /* The parameters. */
00448     jpc_qcxcp_t compparms;
00449 } jpc_qcc_t;
00450
00451 /* QCD marker segment parameters. */
00452
00453 typedef struct {
00454     /* The parameters. */
00455     jpc_qcxcp_t compparms;
00456 } jpc_qcd_t;
00457
00458 /* *****\
00459  * POD marker segment parameters.
00460  * *****/
00461
00462 typedef struct {
00463     /* The progression order. */
00464     uint_fast8_t prgord;
00465
00466     /* The lower bound (inclusive) on the resolution level for the
00467        progression order volume. */
00468     uint_fast8_t rlvlnostart;
00469
00470     /* The upper bound (exclusive) on the resolution level for the
00471        progression order volume. */
00472     uint_fast8_t rlvlnoend;
00473
00474     /* The lower bound (inclusive) on the component for the progression
00475        order volume. */
00476     uint_fast16_t compnostart;

```

```

00484
00485     /* The upper bound (exclusive) on the component for the progression
00486        order volume. */
00487     uint_fast16_t compnoend;
00488
00489     /* The upper bound (exclusive) on the layer for the progression
00490        order volume. */
00491     uint_fast16_t lyrnoend;
00492
00493 } jpc_pocpchg_t;
00494
00495 /* An alias for the above type. */
00496 typedef jpc_pocpchg_t jpc_pchg_t;
00497
00498 /* POC marker segment parameters. */
00499
00500 typedef struct {
00501
00502     /* The number of progression order changes. */
00503     int numpchgs;
00504
00505     /* The per-progression-order-change information. */
00506     jpc_pocpchg_t *pchgs;
00507
00508 } jpc_poc_t;
00509
00510 /*****\
00511 * PPM/PPT marker segment parameters.
00512 \*****/
00513
00514 /* PPM marker segment parameters. */
00515
00516 typedef struct {
00517
00518     /* The index. */
00519     uint_fast8_t ind;
00520
00521     /* The length. */
00522     uint_fast16_t len;
00523
00524     /* The data. */
00525     jas_uchar *data;
00526
00527 } jpc_ppm_t;
00528
00529 /* PPT marker segment parameters. */
00530
00531 typedef struct {
00532
00533     /* The index. */
00534     uint_fast8_t ind;
00535
00536     /* The length. */
00537     uint_fast32_t len;
00538
00539     /* The data. */
00540     unsigned char *data;
00541
00542 } jpc_ppt_t;
00543
00544 /*****\
00545 * COM marker segment parameters.
00546 \*****/
00547
00548 /*
00549  * Registration IDs.
00550  */
00551
00552 #define JPC_COM_BIN          0x00
00553 #define JPC_COM_LATIN       0x01
00554
00555 typedef struct {
00556
00557     /* The registration ID. */
00558     uint_fast16_t regid;
00559
00560     /* The length of the data in bytes. */
00561     uint_fast16_t len;
00562
00563     /* The data. */
00564     jas_uchar *data;

```



```

00565
00566 } jpc_com_t;
00567
00568 /*****
00569  * SOP marker segment parameters.
00570  *****/
00571
00572 typedef struct {
00573
00574     /* The sequence number. */
00575     uint_fast16_t seqno;
00576
00577 } jpc_sop_t;
00578
00579 /*****
00580  * CRG marker segment parameters.
00581  *****/
00582
00583 /* Per component information. */
00584
00585 typedef struct {
00586
00587     /* The horizontal offset. */
00588     uint_fast16_t hoff;
00589
00590     /* The vertical offset. */
00591     uint_fast16_t voff;
00592
00593 } jpc_crgcomp_t;
00594
00595 typedef struct {
00596
00597     /* The number of components. */
00598     int numcomps;
00599
00600     /* Per component information. */
00601     jpc_crgcomp_t *comps;
00602
00603 } jpc_crg_t;
00604
00605 /*****
00606  * Marker segment parameters for unknown marker type.
00607  *****/
00608
00609 typedef struct {
00610
00611     /* The data. */
00612     jas_uchar *data;
00613
00614     /* The length. */
00615     uint_fast16_t len;
00616
00617 } jpc_unk_t;
00618
00619 /*****
00620  * Generic marker segment parameters.
00621  *****/
00622
00623 typedef union {
00624     int soc;           /* unused */
00625     jpc_sot_t sot;
00626     int sod;           /* unused */
00627     int eoc;           /* unused */
00628     jpc_siz_t siz;
00629     jpc_cod_t cod;
00630     jpc_coc_t coc;
00631     jpc_rgn_t rgn;
00632     jpc_qcd_t qcd;
00633     jpc_qcc_t qcc;
00634     jpc_poc_t poc;
00635     /* jpc_plm_t plm; */
00636     /* jpc_plt_t plt; */
00637     jpc_ppm_t ppm;
00638     jpc_ppt_t ppt;
00639     jpc_sop_t sop;
00640     int eph;           /* unused */
00641     jpc_com_t com;
00642     jpc_crg_t crg;
00643     jpc_unk_t unk;
00644 } jpc_msparms_t;
00645

```

```

00646 /*****\
00647 * Marker segment.
00648 \*****/
00649
00650 /* Marker segment IDs. */
00651
00652 /* The smallest valid marker value. */
00653 #define JPC_MS_MIN      0xff00
00654
00655 /* The largest valid marker value. */
00656 #define JPC_MS_MAX      0xffff
00657
00658 /* The minimum marker value that cannot occur within packet data. */
00659 #define JPC_MS_INMIN    0xff80
00660 /* The maximum marker value that cannot occur within packet data. */
00661 #define JPC_MS_INMAX    0xffff
00662
00663 /* Delimiting marker segments. */
00664 #define JPC_MS_SOC      0xff4f /* Start of code stream (SOC). */
00665 #define JPC_MS_SOT      0xff90 /* Start of tile-part (SOT). */
00666 #define JPC_MS_SOD      0xff93 /* Start of data (SOD). */
00667 #define JPC_MS_EOC      0xffd9 /* End of code stream (EOC). */
00668
00669 /* Fixed information marker segments. */
00670 #define JPC_MS_SIZ      0xff51 /* Image and tile size (SIZ). */
00671
00672 /* Functional marker segments. */
00673 #define JPC_MS_COD      0xff52 /* Coding style default (COD). */
00674 #define JPC_MS_COC      0xff53 /* Coding style component (COC). */
00675 #define JPC_MS_RGN      0xff5e /* Region of interest (RGN). */
00676 #define JPC_MS_QCD      0xff5c /* Quantization default (QCD). */
00677 #define JPC_MS_QCC      0xff5d /* Quantization component (QCC). */
00678 #define JPC_MS_POC      0xff5f /* Progression order default (POC). */
00679
00680 /* Pointer marker segments. */
00681 #define JPC_MS_TLM      0xff55 /* Tile-part lengths, main header (TLM). */
00682 #define JPC_MS_PLM      0xff57 /* Packet length, main header (PLM). */
00683 #define JPC_MS_PLT      0xff58 /* Packet length, tile-part header (PLT). */
00684 #define JPC_MS_PPM      0xff60 /* Packed packet headers, main header (PPM). */
00685 #define JPC_MS_PPT      0xff61 /* Packet packet headers, tile-part header (PPT). */
00686
00687 /* In bit stream marker segments. */
00688 #define JPC_MS_SOP      0xff91 /* Start of packet (SOP). */
00689 #define JPC_MS_EPH      0xff92 /* End of packet header (EPH). */
00690
00691 /* Informational marker segments. */
00692 #define JPC_MS_CRG      0xff63 /* Component registration (CRG). */
00693 #define JPC_MS_COM      0xff64 /* Comment (COM). */
00694
00695 /* Forward declaration. */
00696 struct jpc_msops_s;
00697
00698 /* Generic marker segment class. */
00699
00700 typedef struct {
00701
00702     /* The type of marker segment. */
00703     uint_fast16_t id;
00704
00705     /* The length of the marker segment. */
00706     uint_fast16_t len;
00707
00708     /* The starting offset within the stream. */
00709     uint_fast32_t off;
00710
00711     /* The parameters of the marker segment. */
00712     jpc_msparms_t parms;
00713
00714     /* The marker segment operations. */
00715     const struct jpc_msops_s *ops;
00716 } jpc_ms_t;
00717
00718
00719 /* Marker segment operations (which depend on the marker segment type). */
00720
00721 typedef struct jpc_msops_s {
00722
00723     /* Destroy the marker segment parameters. */
00724     void (*destroyparms)(jpc_ms_t *ms);
00725
00726     /* Get the marker segment parameters from a stream. */

```

```

00727     int (*getparms)(jpc_ms_t *ms, jpc_cstate_t *cstate, jas_stream_t *in);
00728
00729     /* Put the marker segment parameters to a stream. */
00730     int (*putparms)(jpc_ms_t *ms, jpc_cstate_t *cstate, jas_stream_t *out);
00731
00732     /* Dump the marker segment parameters (for debugging). */
00733     int (*dumpparms)(jpc_ms_t *ms, FILE *out);
00734
00735 } jpc_msops_t;
00736
00737 /*****
00738  * Macros/Functions.
00739  *****/
00740
00741 /* Create a code-stream state object. */
00742 jpc_cstate_t *jpc_cstate_create(void);
00743
00744 /* Destroy a code-stream state object. */
00745 void jpc_cstate_destroy(jpc_cstate_t *cstate);
00746
00747 /* Create a marker segment. */
00748 jpc_ms_t *jpc_ms_create(int type);
00749
00750 /* Destroy a marker segment. */
00751 void jpc_ms_destroy(jpc_ms_t *ms);
00752
00753 /* Does a marker segment have parameters? */
00754 JAS_ATTRIBUTE_CONST
00755 static inline bool JPC_MS_HASPARMS(unsigned x)
00756 {
00757     return !(x == JPC_MS_SOC || x == JPC_MS_SOD || x == JPC_MS_EOC ||
00758             x == JPC_MS_EPH || (x >= 0xff30 && x <= 0xff3f));
00759 }
00760
00761 /* Get the marker segment type. */
00762 JAS_ATTRIBUTE_PURE
00763 static inline unsigned jpc_ms_gettype(const jpc_ms_t *ms)
00764 {
00765     return ms->id;
00766 }
00767
00768 /* Read a marker segment from a stream. */
00769 jpc_ms_t *jpc_getms(jas_stream_t *in, jpc_cstate_t *cstate);
00770
00771 /* Write a marker segment to a stream. */
00772 int jpc_putms(jas_stream_t *out, jpc_cstate_t *cstate, jpc_ms_t *ms);
00773
00774 /* Copy code stream data from one stream to another. */
00775 int jpc_getdata(jas_stream_t *in, jas_stream_t *out, long n);
00776
00777 /* Copy code stream data from one stream to another. */
00778 int jpc_putdata(jas_stream_t *out, jas_stream_t *in, long n);
00779
00780 /* Dump a marker segment (for debugging). */
00781 void jpc_ms_dump(jpc_ms_t *ms, FILE *out);
00782
00783 /* Read a 8-bit unsigned integer from a stream. */
00784 int jpc_getuint8(jas_stream_t *in, uint_fast8_t *val);
00785
00786 /* Read a 16-bit unsigned integer from a stream. */
00787 int jpc_getuint16(jas_stream_t *in, uint_fast16_t *val);
00788
00789 /* Read a 32-bit unsigned integer from a stream. */
00790 int jpc_getuint32(jas_stream_t *in, uint_fast32_t *val);
00791
00792 /* Write a 8-bit unsigned integer to a stream. */
00793 int jpc_putuint8(jas_stream_t *out, uint_fast8_t val);
00794
00795 /* Write a 16-bit unsigned integer to a stream. */
00796 int jpc_putuint16(jas_stream_t *out, uint_fast16_t val);
00797
00798 /* Write a 32-bit unsigned integer to a stream. */
00799 int jpc_putuint32(jas_stream_t *out, uint_fast32_t val);
00800
00801 #endif

```

## 11.46 jpc\_dec.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * JPEG-2000 Decoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_DEC_H
00071 #define JPC_DEC_H
00072
00073 /*****\
00074  * Includes.
00075  */
00076 /*****\
00077 #include "jasper/jas_stream.h"
00078 #include "jasper/jas_image.h"

```

```

00079
00080 #include "jpc_tsfb.h"
00081 #include "jpc_tagtree.h"
00082 #include "jpc_cs.h"
00083 #include "jpc_t1cod.h"
00084 #include "jpc_t2cod.h"
00085
00086 /*****\
00087  * Below are some ugly warts necessary to support packed packet headers.
00088  \*****/
00089
00090 /* PPM/PPT marker segment table entry. */
00091
00092 typedef struct {
00093
00094     /* The index for this entry. */
00095     uint_fast16_t ind;
00096
00097     /* The data length. */
00098     uint_fast32_t len;
00099
00100     /* The data. */
00101     jas_uchar *data;
00102 } jpc_ppxstabent_t;
00103
00104 /* PPM/PPT marker segment table. */
00105
00106 typedef struct {
00107
00108     /* The number of entries. */
00109     unsigned numents;
00110
00111     /* The maximum number of entries (i.e., the allocated size of the array
00112        below). */
00113     unsigned maxents;
00114
00115     /* The table entries. */
00116     jpc_ppxstabent_t **ents;
00117 } jpc_ppxstab_t;
00118
00119 /* Stream list class. */
00120
00121 typedef struct {
00122
00123     /* The number of streams in this list. */
00124     unsigned numstreams;
00125
00126     /* The maximum number of streams that can be accomodated without
00127        growing the streams array. */
00128     unsigned maxstreams;
00129
00130     /* The streams. */
00131     jas_stream_t **streams;
00132 } jpc_streamlist_t;
00133
00134 /*****\
00135  * Coding parameters class.
00136  \*****/
00137
00138 /* Per-component coding parameters. */
00139
00140 typedef struct {
00141
00142     /* How were various coding parameters set? */
00143     unsigned flags;
00144
00145     /* Per-component coding style parameters (e.g., explicit precinct sizes) */
00146     uint_fast8_t csty;
00147
00148     /* The number of resolution levels. */
00149     uint_fast8_t numrlvls;
00150
00151     /* The code block width exponent. */
00152     uint_fast8_t cblkwidthexpn;
00153
00154     /* The code block height exponent. */
00155     uint_fast8_t cblkheightexpn;
00156 }
00157
00158
00159

```

```

00160      /* The QMFB ID. */
00161      uint_fast8_t qmfbid;
00162
00163      /* The quantization style. */
00164      uint_fast8_t qsty;
00165
00166      /* The number of quantizer step sizes. */
00167      uint_fast16_t numstepsizes;
00168
00169      /* The step sizes. */
00170      uint_fast16_t stepsizes[3 * JPC_MAXRLVLS + 1];
00171
00172      /* The number of guard bits. */
00173      uint_fast8_t numguardbits;
00174
00175      /* The ROI shift value. */
00176      uint_fast8_t roishift;
00177
00178      /* The code block parameters. */
00179      uint_fast8_t cblkctx;
00180
00181      /* The precinct width exponents. */
00182      uint_fast8_t prcwidthexpns[JPC_MAXRLVLS];
00183
00184      /* The precinct height exponents. */
00185      uint_fast8_t prcheightexpns[JPC_MAXRLVLS];
00186
00187 } jpc_dec_ccp_t;
00188
00189 /* Coding parameters. */
00190
00191 typedef struct {
00192
00193     /* How were these coding parameters set? */
00194     unsigned flags;
00195
00196     /* Progression change list. */
00197     jpc_pchglist_t *pchglist;
00198
00199     /* Progression order. */
00200     uint_fast8_t prgord;
00201
00202     /* The number of layers. */
00203     uint_fast16_t numlyrs;
00204
00205     /* The MCT ID. */
00206     uint_fast8_t mctid;
00207
00208     /* The coding style parameters (e.g., SOP, EPH). */
00209     uint_fast8_t csty;
00210
00211     /* The number of components. */
00212     unsigned numcomps;
00213
00214     /* The per-component coding parameters. */
00215     jpc_dec_ccp_t *ccps;
00216
00217 } jpc_dec_cp_t;
00218
00219 /*****\
00220 * Decoder class.
00221 \*****/
00222
00223 /* Decoder per-segment state information. */
00224
00225 typedef struct jpc_dec_seg_s {
00226
00227     /* The next segment in the list. */
00228     struct jpc_dec_seg_s *next;
00229
00230     /* The previous segment in the list. */
00231     struct jpc_dec_seg_s *prev;
00232
00233     /* The starting pass number for this segment. */
00234     unsigned passno;
00235
00236     /* The number of passes in this segment. */
00237     unsigned numpasses;
00238
00239     /* The maximum number of passes in this segment. */
00240     unsigned maxpasses;

```

```

00241
00242     /* The type of data in this segment (i.e., MQ or raw). */
00243     enum jpc_segtype type;
00244
00245     /* A stream containing the data for this segment. */
00246     jas_stream_t *stream;
00247
00248     /* The number of bytes destined for this segment from the packet
00249        currently being decoded. */
00250     unsigned cnt;
00251
00252     /* A flag indicating if this segment has been terminated. */
00253     int complete;
00254
00255     /* The layer number to which this segment belongs. */
00256     /* If the segment spans multiple layers, then the largest layer number
00257        spanned by the segment is used. */
00258     unsigned lyrno;
00259 } jpc_dec_seg_t;
00260
00261 /* Decoder segment list. */
00262
00263 typedef struct {
00264     /* The first entry in the list. */
00265     jpc_dec_seg_t *head;
00266
00267     /* The last entry in the list. */
00268     jpc_dec_seg_t *tail;
00269 } jpc_dec_seglist_t;
00270
00271 /* Decoder per-code-block state information. */
00272
00273 typedef struct {
00274     /* The number of passes. */
00275     unsigned numpasses;
00276
00277     /* A list of segments that still need to be decoded. */
00278     jpc_dec_seglist_t segs;
00279
00280     /* The first incomplete/partial segment. */
00281     jpc_dec_seg_t *curseg;
00282
00283     /* The number of leading insignificant bit planes for this code block. */
00284     unsigned numimsbs;
00285
00286     /* The number of bits used to encode pass data lengths. */
00287     unsigned numlenbits;
00288
00289     /* The first pass number containing data for this code block. */
00290     unsigned firstpassno;
00291
00292     /* The sample data associated with this code block. */
00293     jas_matrix_t *data;
00294 } jpc_dec_cblk_t;
00295
00296 /* Decoder per-code-block-group state information. */
00297
00298 typedef struct {
00299     /* The x-coordinate of the top-left corner of the precinct. */
00300     uint_fast32_t xstart;
00301
00302     /* The y-coordinate of the top-left corner of the precinct. */
00303     uint_fast32_t ystart;
00304
00305     /* The x-coordinate of the bottom-right corner of the precinct
00306        (plus one). */
00307     uint_fast32_t xend;
00308
00309     /* The y-coordinate of the bottom-right corner of the precinct
00310        (plus one). */
00311     uint_fast32_t yend;
00312
00313     /* The number of code blocks spanning this precinct in the horizontal
00314        direction. */
00315     unsigned numhcbks;

```

```

00322
00323     /* The number of code blocks spanning this precinct in the vertical
00324     direction. */
00325     unsigned numvcbks;
00326
00327     /* The total number of code blocks in this precinct. */
00328     unsigned numcblks;
00329
00330     /* The per code block information. */
00331     jpc_dec_cblk_t *cblks;
00332
00333     /* The inclusion tag tree. */
00334     jpc_tagtree_t *incltagtree;
00335
00336     /* The insignificant MSBs tag tree. */
00337     jpc_tagtree_t *numimbsstagtree;
00338
00339 } jpc_dec_prc_t;
00340
00341 /* Decoder per-band state information. */
00342
00343 typedef struct {
00344
00345     /* The per-code-block-group state information. */
00346     jpc_dec_prc_t *prcs;
00347
00348     /* The sample data associated with this band. */
00349     jas_matrix_t *data;
00350
00351     /* The orientation of this band (i.e., LL, LH, HL, or HH). */
00352     enum jpc_tsfb_orient orient;
00353
00354     /* The encoded quantizer step size. */
00355     unsigned stepsize;
00356
00357     /* The absolute quantizer step size. */
00358     jpc_fix_t absstepsize;
00359
00360     /* The number of bit planes for this band. */
00361     unsigned numbps;
00362
00363     /* The analysis gain associated with this band. */
00364     int analgain;
00365
00366     /* The ROI shift value for this band. */
00367     int roishift;
00368
00369 } jpc_dec_band_t;
00370
00371 /* Decoder per-resolution-level state information. */
00372
00373 typedef struct {
00374
00375     /* The number of bands associated with this resolution level. */
00376     unsigned numbands;
00377
00378     /* The per-band information. */
00379     jpc_dec_band_t *bands;
00380
00381     /* The x-coordinate of the top-left corner of the tile-component
00382     at this resolution. */
00383     uint_fast32_t xstart;
00384
00385     /* The y-coordinate of the top-left corner of the tile-component
00386     at this resolution. */
00387     uint_fast32_t ystart;
00388
00389     /* The x-coordinate of the bottom-right corner of the tile-component
00390     at this resolution (plus one). */
00391     uint_fast32_t xend;
00392
00393     /* The y-coordinate of the bottom-right corner of the tile-component
00394     at this resolution (plus one). */
00395     uint_fast32_t yend;
00396
00397     /* The exponent value for the nominal precinct width measured
00398     relative to the associated LL band. */
00399     unsigned prcwidthhexpn;
00400
00401     /* The exponent value for the nominal precinct height measured
00402     relative to the associated LL band. */

```



```

00403     unsigned prcheightexpn;
00404
00405     /* The number of precincts in the horizontal direction. */
00406     unsigned numhprcs;
00407
00408     /* The number of precincts in the vertical direction. */
00409     unsigned numvprcs;
00410
00411     /* The total number of precincts. */
00412     unsigned numprcs;
00413
00414     /* The exponent value for the nominal code block group width.
00415      * This quantity is associated with the next lower resolution level
00416      * (assuming that there is one). */
00417     unsigned cbgwidthexpn;
00418
00419     /* The exponent value for the nominal code block group height
00420      * This quantity is associated with the next lower resolution level
00421      * (assuming that there is one). */
00422     unsigned cbgheightexpn;
00423
00424     /* The exponent value for the code block width. */
00425     uint_fast16_t cblkwidthexpn;
00426
00427     /* The exponent value for the code block height. */
00428     uint_fast16_t cblkheightexpn;
00429
00430 } jpc_dec_rlvl_t;
00431
00432 /* Decoder per-tile-component state information. */
00433
00434 typedef struct {
00435
00436     /* The x-coordinate of the top-left corner of the tile-component
00437      * in the coordinate system of the tile-component. */
00438     uint_fast32_t xstart;
00439
00440     /* The y-coordinate of the top-left corner of the tile-component
00441      * in the coordinate system of the tile-component. */
00442     uint_fast32_t ystart;
00443
00444     /* The x-coordinate of the bottom-right corner of the tile-component
00445      * in the coordinate system of the tile-component (plus one). */
00446     uint_fast32_t xend;
00447
00448     /* The y-coordinate of the bottom-right corner of the tile-component
00449      * in the coordinate system of the tile-component (plus one). */
00450     uint_fast32_t yend;
00451
00452     /* The component data for the current tile. */
00453     jas_matrix_t *data;
00454
00455     /* The number of resolution levels. */
00456     unsigned numrlvls;
00457
00458     /* The per resolution level information. */
00459     jpc_dec_rlvl_t *rlvls;
00460
00461     /* The TSFB. */
00462     jpc_tsfb_t *tsfb;
00463
00464 } jpc_dec_tcomp_t;
00465
00466 /*
00467  * Tile states.
00468  */
00469
00470 #define JPC_TILE_INIT    0
00471 #define JPC_TILE_ACTIVE 1
00472 #define JPC_TILE_ACTIVELAST 2
00473 #define JPC_TILE_DONE    3
00474
00475 /* Decoder per-tile state information. */
00476
00477 typedef struct {
00478
00479     /* The processing state for this tile. */
00480     int state;
00481
00482     /* The x-coordinate of the top-left corner of the tile on the reference
00483      * grid. */

```

```

00484     uint_fast32_t xstart;
00485
00486     /* The y-coordinate of the top-left corner of the tile on the reference
00487        grid. */
00488     uint_fast32_t ystart;
00489
00490     /* The x-coordinate of the bottom-right corner of the tile on the
00491        reference grid (plus one). */
00492     uint_fast32_t xend;
00493
00494     /* The y-coordinate of the bottom-right corner of the tile on the
00495        reference grid (plus one). */
00496     uint_fast32_t yend;
00497
00498     /* The packed packet header data for this tile. */
00499     jpc_ppxstab_t *pptstab;
00500
00501     /* A stream containing the packed packet header data for this tile. */
00502     jas_stream_t *pkthdrstream;
00503
00504     /* The coding parameters for this tile. */
00505     jpc_dec_cp_t *cp;
00506
00507     /* The per tile-component information. */
00508     jpc_dec_tcomp_t *tcomps;
00509
00510     /* The next expected tile-part number. */
00511     unsigned partno;
00512
00513     /* The number of tile-parts. */
00514     unsigned numparts;
00515
00516     /* The coding mode. */
00517     int realmode;
00518
00519     /* The packet iterator for this tile. */
00520     jpc_pi_t *pi;
00521
00522 } jpc_dec_tile_t;
00523
00524 /* Decoder per-component state information. */
00525
00526 typedef struct {
00527
00528     /* The horizontal sampling period. */
00529     uint_fast32_t hstep;
00530
00531     /* The vertical sampling period. */
00532     uint_fast32_t vstep;
00533
00534     /* The number of samples in the horizontal direction. */
00535     uint_fast32_t width;
00536
00537     /* The number of samples in the vertical direction. */
00538     uint_fast32_t height;
00539
00540     /* The precision of the sample data. */
00541     uint_fast16_t prec;
00542
00543     /* The signedness of the sample data. */
00544     bool sgnd;
00545
00546     /* The sample alignment horizontal offset. */
00547     uint_fast32_t hsubstep;
00548
00549     /* The sample alignment vertical offset. */
00550     uint_fast32_t vsubstep;
00551
00552 } jpc_dec_cmpt_t;
00553
00554 /* Decoder state information. */
00555
00556 typedef struct {
00557
00558     /* The decoded image. */
00559     jas_image_t *image;
00560
00561     /* The x-coordinate of the top-left corner of the image area on
00562        the reference grid. */
00563     uint_fast32_t xstart;
00564

```

```
00565     /* The y-coordinate of the top-left corner of the image area on
00566        the reference grid. */
00567     uint_fast32_t ystart;
00568
00569     /* The x-coordinate of the bottom-right corner of the image area on
00570        the reference grid (plus one). */
00571     uint_fast32_t xend;
00572
00573     /* The y-coordinate of the bottom-right corner of the image area on
00574        the reference grid (plus one). */
00575     uint_fast32_t yend;
00576
00577     /* The nominal tile width in units of the image reference grid. */
00578     uint_fast32_t tilewidth;
00579
00580     /* The nominal tile height in units of the image reference grid. */
00581     uint_fast32_t tileheight;
00582
00583     /* The horizontal offset from the origin of the reference grid to the
00584        left side of the first tile. */
00585     uint_fast32_t tilexoff;
00586
00587     /* The vertical offset from the origin of the reference grid to the
00588        top side of the first tile. */
00589     uint_fast32_t tileyoff;
00590
00591     /* The number of tiles spanning the image area in the vertical
00592        direction. */
00593     unsigned numhtiles;
00594
00595     /* The number of tiles spanning the image area in the horizontal
00596        direction. */
00597     unsigned numvtiles;
00598
00599     /* The total number of tiles. */
00600     unsigned numtiles;
00601
00602     /* The per-tile information. */
00603     jpc_dec_tile_t *tiles;
00604
00605     /* The tile currently being processed. */
00606     jpc_dec_tile_t *curtile;
00607
00608     /* The number of components. */
00609     unsigned numcomps;
00610
00611     /* The stream containing the input JPEG-2000 code stream data. */
00612     jas_stream_t *in;
00613
00614     /* The default coding parameters for all tiles. */
00615     jpc_dec_cp_t *cp;
00616
00617     /* The maximum number of layers that may be decoded. */
00618     unsigned maxlyrs;
00619
00620     /* The maximum number of packets that may be decoded. */
00621     int maxpkts;
00622
00623     /* The number of packets decoded so far in the processing of the entire
00624        code stream. */
00625     unsigned numpkts;
00626
00627     /* The next expected PPM marker segment sequence number. */
00628     unsigned ppmseqno;
00629
00630     /* The current state for code stream processing. */
00631     int state;
00632
00633     /* The per-component information. */
00634     jpc_dec_cmpt_t *cmpts;
00635
00636     /* The information from PPM marker segments. */
00637     jpc_ppxstab_t *ppmstab;
00638
00639     /* A list of streams containing packet header data from PPM marker
00640        segments. */
00641     jpc_streamlist_t *pkthdrstreams;
00642
00643     /* The expected ending offset for a tile-part. */
00644     long curtileendoff;
00645
```

```

00646      /* This is required by the tier-2 decoder. */
00647      jpc_cstate_t *cstate;
00648
00649      size_t max_samples;
00650
00651 } jpc_dec_t;
00652
00653 /* Decoder options. */
00654
00655 typedef struct {
00656
00657     /* The debug level for the decoder. */
00658     int debug;
00659
00660     /* The maximum number of layers to decode. */
00661     unsigned maxlyrs;
00662
00663     /* The maximum number of packets to decode. */
00664     int maxpkts;
00665
00666     size_t max_samples;
00667
00668 } jpc_dec_importopts_t;
00669
00670 /*****
00671  * Functions.
00672  \*****/
00673
00674 /* Create a decoder segment object. */
00675 jpc_dec_seg_t *jpc_seg_alloc(void);
00676
00677 /* Destroy a decoder segment object. */
00678 void jpc_seg_destroy(jpc_dec_seg_t *seg);
00679
00680 /* Remove a segment from a segment list. */
00681 void jpc_seglist_remove(jpc_dec_seglist_t *list, jpc_dec_seg_t *node);
00682
00683 /* Insert a segment into a segment list. */
00684 void jpc_seglist_insert(jpc_dec_seglist_t *list, jpc_dec_seg_t *ins,
00685     jpc_dec_seg_t *node);
00686
00687 #endif

```

## 11.47 jpc\_enc.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.

```

```

00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065 * $Id$
00066 */
00067
00068 #ifndef JPC_ENC_H
00069 #define JPC_ENC_H
00070
00071 /*****\
00072 * Includes.
00073 \*****/
00074
00075 #include "jasper/jas_image.h"
00076 #include "jasper/jas_seq.h"
00077 #include "jasper/jas_stream.h"
00078
00079 #include "jpc_t1cod.h"
00080 #include "jpc_t2cod.h"
00081 #include "jpc_mqenc.h"
00082 #include "jpc_tagtree.h"
00083 #include "jpc_cs.h"
00084 #include "jpc_fix.h"
00085 #include "jpcflt.h"
00086 #include "jpc_tsfb.h"
00087
00088 /*****\
00089 * Constants.
00090 \*****/
00091
00092 /* The number of bits used in various lookup tables. */
00093 #define JPC_NUMEXTRABITS JPC_NMSEDEC_FRACBITS
00094
00095 /* An invalid R-D slope value. */
00096 #define JPC_BADRDSLOPE (-1)
00097
00098 /*****\
00099 * Coding parameters types.
00100 \*****/
00101
00102 /* Per-component coding paramters. */
00103
00104 typedef struct {
00105
00106     /* The horizontal sampling period. */
00107     uint_fast8_t sampgrdstpx;
00108
00109     /* The vertical sampling period. */
00110     uint_fast8_t sampgrdstpy;
00111
00112     /* The sample alignment horizontal offset. */
00113     uint_fast8_t sampgrdsbstpx;

```

```

00114
00115     /* The sample alignment vertical offset. */
00116     uint_fast8_t sampgrdsbstepy;
00117
00118     /* The precision of the samples. */
00119     uint_fast8_t prec;
00120
00121     /* The signedness of the samples. */
00122     bool sgnd;
00123
00124     /* The number of step sizes. */
00125     uint_fast16_t numstepsizes;
00126
00127     /* The quantizer step sizes. */
00128     uint_fast16_t stepsizes[JPC_MAXBANDS];
00129
00130 } jpc_enc_ccp_t;
00131
00132 /* Per-tile coding parameters. */
00133
00134 typedef struct {
00135     /* The coding mode. */
00136     bool intmode;
00137
00138     /* The coding style (i.e., SOP, EPH). */
00139     uint_fast8_t csty;
00140
00141     /* The progression order. */
00142     uint_fast8_t prg;
00143
00144     /* The multicomponent transform. */
00145     uint_fast8_t mctid;
00146
00147     /* The number of layers. */
00148     uint_fast16_t numlyrs;
00149
00150     /* The normalized bit rates associated with the various
00151        intermediate layers. */
00152     jpc_fix_t *ilyrrates;
00153 } jpc_enc_tcp_t;
00154
00155 /* Per tile-component coding parameters. */
00156
00157 typedef struct {
00158     /* The coding style (i.e., explicit precinct sizes). */
00159     uint_fast8_t csty;
00160
00161     /* The maximum number of resolution levels allowed. */
00162     uint_fast8_t maxrlvls;
00163
00164     /* The exponent for the nominal code block width. */
00165     uint_fast16_t cblkwidthexpn;
00166
00167     /* The exponent for the nominal code block height. */
00168     uint_fast16_t cblkheightexpn;
00169
00170     /* The code block style parameters (e.g., lazy, terminate all,
00171        segmentation symbols, causal, reset probability models). */
00172     uint_fast8_t cblksty;
00173
00174     /* The QMFB. */
00175     uint_fast8_t qmfbid;
00176
00177     /* The precinct width values. */
00178     uint_fast16_t prcwidthexpns[JPC_MAXRLVLS];
00179
00180     /* The precinct height values. */
00181     uint_fast16_t prcheightexpns[JPC_MAXRLVLS];
00182
00183     /* The number of guard bits. */
00184     uint_fast8_t numgbits;
00185 } jpc_enc_tccp_t;
00186
00187 /* Coding parameters. */
00188
00189 typedef struct {
00190

```

```

00195     /* The debug level. */
00196     int debug;
00197
00198     /* The horizontal offset from the origin of the reference grid to the
00199        left edge of the image area. */
00200     uint_fast32_t imgareatlx;
00201
00202     /* The vertical offset from the origin of the reference grid to the
00203        top edge of the image area. */
00204     uint_fast32_t imgareatly;
00205
00206     /* The horizontal offset from the origin of the reference grid to the
00207        right edge of the image area (plus one). */
00208     uint_fast32_t refgrdwidth;
00209
00210     /* The vertical offset from the origin of the reference grid to the
00211        bottom edge of the image area (plus one). */
00212     uint_fast32_t refgrdheight;
00213
00214     /* The horizontal offset from the origin of the tile grid to the
00215        origin of the reference grid. */
00216     uint_fast32_t tilegrdoffx;
00217
00218     /* The vertical offset from the origin of the tile grid to the
00219        origin of the reference grid. */
00220     uint_fast32_t tilegrdoffy;
00221
00222     /* The nominal tile width in units of the image reference grid. */
00223     uint_fast32_t tilewidth;
00224
00225     /* The nominal tile height in units of the image reference grid. */
00226     uint_fast32_t tileheight;
00227
00228     /* The number of tiles spanning the image area in the horizontal
00229        direction. */
00230     uint_fast32_t numhtiles;
00231
00232     /* The number of tiles spanning the image area in the vertical
00233        direction. */
00234     uint_fast32_t numvtiles;
00235
00236     /* The number of tiles. */
00237     uint_fast32_t numtiles;
00238
00239     /* The number of components. */
00240     uint_fast16_t numcmpts;
00241
00242     /* The per-component coding parameters. */
00243     jpc_enc_ccp_t *ccps;
00244
00245     /* The per-tile coding parameters. */
00246     jpc_enc_tcp_t tcp;
00247
00248     /* The per-tile-component coding parameters. */
00249     jpc_enc_tccp_t tccp;
00250
00251     /* The target code stream length in bytes. */
00252     uint_fast32_t totalsize;
00253
00254     /* The raw (i.e., uncompressed) size of the image in bytes. */
00255     uint_fast32_t rawsize;
00256
00257 } jpc_enc_cp_t;
00258
00259 /*****
00260  * Encoder class.
00261  *****/
00262
00263 /* Encoder per-coding-pass state information. */
00264
00265 typedef struct {
00266
00267     /* The starting offset for this pass. */
00268     int start;
00269
00270     /* The ending offset for this pass. */
00271     int end;
00272
00273     /* The type of data in this pass (i.e., MQ or raw). */
00274     enum jpc_segtype type;
00275

```

```

00276      /* Flag indicating that this pass is terminated. */
00277      int term;
00278
00279      /* The entropy coder state after coding this pass. */
00280      jpc_mqencstate_t mqencstate;
00281
00282      /* The layer to which this pass has been assigned. */
00283      unsigned lyrno;
00284
00285      /* The R-D slope for this pass. */
00286      jpc_flt_t rdslope;
00287
00288      /* The weighted MSE reduction associated with this pass. */
00289      jpc_flt_t wmsedec;
00290
00291      /* The cumulative weighted MSE reduction. */
00292      jpc_flt_t cumwmsedec;
00293
00294      /* The normalized MSE reduction. */
00295      long nmsedec;
00296
00297 } jpc_enc_pass_t;
00298
00299 /* Encoder per-code-block state information. */
00300
00301 typedef struct {
00302
00303     /* The number of passes. */
00304     unsigned numpasses;
00305
00306     /* The per-pass information. */
00307     jpc_enc_pass_t *passes;
00308
00309     /* The number of passes encoded so far. */
00310     int numencpasses;
00311
00312     /* The number of insignificant MSBs. */
00313     int numimsbs;
00314
00315     /* The number of bits used to encode pass data lengths. */
00316     int numlenbits;
00317
00318     /* The byte stream for this code block. */
00319     jas_stream_t *stream;
00320
00321     /* The entropy encoder. */
00322     jpc_mqenc_t *mqenc;
00323
00324     /* The data for this code block. */
00325     jas_matrix_t *data;
00326
00327     /* The state for this code block. */
00328     jas_matrix_t *flags;
00329
00330     /* The number of bit planes required for this code block. */
00331     int numbps;
00332
00333     /* The next pass to be encoded. */
00334     jpc_enc_pass_t *curpass;
00335
00336     /* The per-code-block-group state information. */
00337     struct jpc_enc_prc_s *prc;
00338
00339     /* The saved current pass. */
00340     /* This is used by the rate control code. */
00341     jpc_enc_pass_t *savedcurpass;
00342
00343     /* The saved length indicator size. */
00344     /* This is used by the rate control code. */
00345     int savednumlenbits;
00346
00347     /* The saved number of encoded passes. */
00348     /* This is used by the rate control code. */
00349     int savednumencpasses;
00350
00351 } jpc_enc_cblk_t;
00352
00353 /* Encoder per-code-block-group state information. */
00354
00355 typedef struct jpc_enc_prc_s {
00356

```



```

00357     /* The x-coordinate of the top-left corner of the precinct. */
00358     uint_fast32_t tlx;
00359
00360     /* The y-coordinate of the top-left corner of the precinct. */
00361     uint_fast32_t tly;
00362
00363     /* The x-coordinate of the bottom-right corner of the precinct
00364        (plus one). */
00365     uint_fast32_t brx;
00366
00367     /* The y-coordinate of the bottom-right corner of the precinct
00368        (plus one). */
00369     uint_fast32_t bry;
00370
00371     /* The number of code blocks spanning the precinct in the horizontal
00372        direction. */
00373     int numhcbks;
00374
00375     /* The number of code blocks spanning the precinct in the vertical
00376        direction. */
00377     int numvcblks;
00378
00379     /* The total number of code blocks. */
00380     unsigned numcblks;
00381
00382     /* The per-code-block information. */
00383     jpc_enc_cblk_t *cblks;
00384
00385     /* The inclusion tag tree. */
00386     jpc_tagtree_t *incltree;
00387
00388     /* The insignificant MSBs tag tree. */
00389     jpc_tagtree_t *nlibtree;
00390
00391     /* The per-band information. */
00392     struct jpc_enc_band_s *band;
00393
00394     /* The saved inclusion tag tree. */
00395     /* This is used by rate control. */
00396     jpc_tagtree_t *savincltree;
00397
00398     /* The saved leading-insignificant-bit-planes tag tree. */
00399     /* This is used by rate control. */
00400     jpc_tagtree_t *savnlibtree;
00401
00402 } jpc_enc_prc_t;
00403
00404 /* Encoder per-band state information. */
00405
00406 typedef struct jpc_enc_band_s {
00407
00408     /* The per precinct information. */
00409     jpc_enc_prc_t *prcs;
00410
00411     /* The coefficient data for this band. */
00412     jas_matrix_t *data;
00413
00414     /* The orientation of this band (i.e., LL, LH, HL, or HH). */
00415     enum jpc_tsfb_orient orient;
00416
00417     /* The number of bit planes associated with this band. */
00418     int numbps;
00419
00420     /* The quantizer step size. */
00421     jpc_fix_t absstepsize;
00422
00423     /* The encoded quantizer step size. */
00424     int stepsize;
00425
00426     /* The L2 norm of the synthesis basis functions associated with
00427        this band. (The MCT is not considered in this value.) */
00428     jpc_fix_t synweight;
00429
00430     /* The analysis gain for this band. */
00431     int analgain;
00432
00433     /* The per-resolution-level information. */
00434     struct jpc_enc_rlvl_s *rlvl;
00435
00436 } jpc_enc_band_t;
00437

```

```

00438 /* Encoder per-resolution-level state information. */
00439
00440 typedef struct jpc_enc_rlvl_s {
00441
00442     /* The x-coordinate of the top-left corner of the tile-component
00443        at this resolution. */
00444     uint_fast32_t tlx;
00445
00446     /* The y-coordinate of the top-left corner of the tile-component
00447        at this resolution. */
00448     uint_fast32_t tly;
00449
00450     /* The x-coordinate of the bottom-right corner of the tile-component
00451        at this resolution (plus one). */
00452     uint_fast32_t brx;
00453
00454     /* The y-coordinate of the bottom-right corner of the tile-component
00455        at this resolution (plus one). */
00456     uint_fast32_t bry;
00457
00458     /* The exponent value for the nominal precinct width measured
00459        relative to the associated LL band. */
00460     int prcwidthexpn;
00461
00462     /* The exponent value for the nominal precinct height measured
00463        relative to the associated LL band. */
00464     int prcheightexpn;
00465
00466     /* The number of precincts spanning the resolution level in the
00467        horizontal direction. */
00468     int numhprcs;
00469
00470     /* The number of precincts spanning the resolution level in the
00471        vertical direction. */
00472     int numvprcs;
00473
00474     /* The total number of precincts. */
00475     unsigned numprcs;
00476
00477     /* The exponent value for the nominal code block group width.
00478        This quantity is associated with the next lower resolution level
00479        (assuming that there is one). */
00480     unsigned cbgwidthexpn;
00481
00482     /* The exponent value for the nominal code block group height.
00483        This quantity is associated with the next lower resolution level
00484        (assuming that there is one). */
00485     unsigned cbgheightexpn;
00486
00487     /* The exponent value for the code block width. */
00488     uint_fast16_t cblkwidthexpn;
00489
00490     /* The exponent value for the code block height. */
00491     uint_fast16_t cblkheightexpn;
00492
00493     /* The number of bands associated with this resolution level. */
00494     unsigned numbands;
00495
00496     /* The per-band information. */
00497     jpc_enc_band_t *bands;
00498
00499     /* The parent tile-component. */
00500     struct jpc_enc_tcmtpt_s *tcmtpt;
00501 } jpc_enc_rlvl_t;
00502
00503 /* Encoder per-tile-component state information. */
00504
00505 typedef struct jpc_enc_tcmtpt_s {
00506
00507     /* The number of resolution levels. */
00508     unsigned numrlvls;
00509
00510     /* The per-resolution-level information. */
00511     jpc_enc_rlvl_t *rlvls;
00512
00513     /* The tile-component data. */
00514     jas_matrix_t *data;
00515
00516     /* The QMFB. */
00517     int qmfbid;

```

```

00519
00520     /* The number of bands. */
00521     int numbands;
00522
00523     /* The TSFB. */
00524     jpc_tsfb_t *tsfb;
00525
00526     /* The synthesis energy weight (for the MCT). */
00527     jpc_fix_t synweight;
00528
00529     /* The precinct width exponents. */
00530     int prcwidthexpns[JPC_MAXRLVLS];
00531
00532     /* The precinct height exponents. */
00533     int prcheightexpns[JPC_MAXRLVLS];
00534
00535     /* The code block width exponent. */
00536     int blkwidthexpn;
00537
00538     /* The code block height exponent. */
00539     int blkheightexpn;
00540
00541     /* Coding style (i.e., explicit precinct sizes). */
00542     int csty;
00543
00544     /* Code block style. */
00545     int blksty;
00546
00547     /* The number of quantizer step sizes. */
00548     int numstepsizes;
00549
00550     /* The encoded quantizer step sizes. */
00551     uint_fast16_t stepsizes[JPC_MAXBANDS];
00552
00553     /* The parent tile. */
00554     struct jpc_enc_tile_s *tile;
00555
00556 } jpc_enc_tcmpt_t;
00557
00558 /* Encoder per-tile state information. */
00559 typedef struct jpc_enc_tile_s {
00560
00561     /* The tile number. */
00562     uint_fast32_t tileno;
00563
00564     /* The x-coordinate of the top-left corner of the tile measured with
00565        respect to the reference grid. */
00566     uint_fast32_t tlx;
00567
00568     /* The y-coordinate of the top-left corner of the tile measured with
00569        respect to the reference grid. */
00570     uint_fast32_t tly;
00571
00572     /* The x-coordinate of the bottom-right corner of the tile measured
00573        with respect to the reference grid (plus one). */
00574     uint_fast32_t brx;
00575
00576     /* The y-coordinate of the bottom-right corner of the tile measured
00577        with respect to the reference grid (plus one). */
00578     uint_fast32_t bry;
00579
00580     /* The coding style. */
00581     uint_fast8_t csty;
00582
00583     /* The progression order. */
00584     uint_fast8_t prg;
00585
00586     /* The number of layers. */
00587     unsigned numlyrs;
00588
00589     /* The MCT to employ (if any). */
00590     uint_fast8_t mctid;
00591
00592     /* The packet iterator (used to determine the order of packet
00593        generation). */
00594     jpc_pi_t *pi;
00595
00596     /* The coding mode (i.e., integer or real). */
00597     bool intmode;
00598
00599

```

```

00600      /* The number of bytes to allocate to the various layers. */
00601      uint_fast32_t *lyrsizes;
00602
00603      /* The number of tile-components. */
00604      unsigned numtcmts;
00605
00606      /* The per tile-component information. */
00607      jpc_enc_tcmpt_t *tcmts;
00608
00609      /* The raw (i.e., uncompressed) size of this tile. */
00610      uint_fast32_t rawsize;
00611
00612 } jpc_enc_tile_t;
00613
00614 /* Encoder class. */
00615
00616 typedef struct jpc_enc_s {
00617
00618     /* The image being encoded. */
00619     jas_image_t *image;
00620
00621     /* The output stream. */
00622     jas_stream_t *out;
00623
00624     /* The coding parameters. */
00625     jpc_enc_cp_t *cp;
00626
00627     /* The tile currently being processed. */
00628     jpc_enc_tile_t *curtile;
00629
00630     /* The code stream state. */
00631     jpc_cstate_t *cstate;
00632
00633     /* The number of bytes output so far. */
00634     uint_fast32_t len;
00635
00636     /* The number of bytes available for the main body of the code stream. */
00637     /* This is used for rate allocation purposes. */
00638     uint_fast32_t mainbodysize;
00639
00640     /* The marker segment currently being processed. */
00641     /* This member is a convenience for making cleanup easier. */
00642     jpc_ms_t *mrk;
00643
00644     /* The stream used to temporarily hold tile-part data. */
00645     jas_stream_t *tmpstream;
00646
00647 } jpc_enc_t;
00648
00649 #endif

```

## 11.48 jpc\_fix.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * Jasper License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:

```

```

00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065  * Fixed-Point Number Class
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_FIX_H
00071 #define JPC_FIX_H
00072
00073 /*****\
00074  * Includes.
00075  \*****/
00076
00077 #include "jasper/jas_config.h"
00078 #include "jasper/jas_types.h"
00079 #include "jasper/jas_fix.h"
00080 #include "jasper/jas_math.h"
00081
00082 /*****\
00083  * Basic parameters of the fixed-point type.
00084  \*****/
00085
00086 /* The integral type used to represent a fixed-point number. This
00087  * type must be capable of representing values from  $-(2^{31})$  to  $2^{31}-1$ 
00088  * (inclusive). */
00089 #ifdef JAS_ENABLE_32BIT
00090 typedef int_least32_t jpc_fix_t;
00091 #else
00092 typedef int_fast32_t jpc_fix_t;
00093 #endif
00094
00095 /* The integral type used to represent higher-precision intermediate results.
00096  * This type should be capable of representing values from  $-(2^{63})$  to  $2^{63}-1$ 
00097  * (inclusive). */
00098 typedef int_fast64_t jpc_fix_big_t;
00099
00100 /* The number of bits used for the fractional part of a fixed-point number. */
00101 #define JPC_FIX_FRACBITS 13
00102
00103 /*****\
00104  * Instantiations of the generic fixed-point number macros for the
00105  * parameters given above. (Too bad C does not support templates, eh?)

```

```

00106 * The purpose of these macros is self-evident if one examines the
00107 * corresponding macros in the jasper/jas_fix.h header file.
00108 \*****
00109
00110 #define JPC_FIX_ZERO      JAS_FIX_ZERO(jpc_fix_t, JPC_FIX_FRACBITS)
00111 #define JPC_FIX_ONE      JAS_FIX_ONE(jpc_fix_t, JPC_FIX_FRACBITS)
00112 #define JPC_FIX_HALF     JAS_FIX_HALF(jpc_fix_t, JPC_FIX_FRACBITS)
00113
00114 JAS_ATTRIBUTE_CONST
00115 static inline jpc_fix_t jpc_inttofix(int x)
00116 {
00117     return JAS_INTTOFIX(jpc_fix_t, JPC_FIX_FRACBITS, x);
00118 }
00119
00120 JAS_ATTRIBUTE_CONST
00121 static inline int jpc_fixtoint(jpc_fix_t x)
00122 {
00123     return JAS_FIXTOINT(jpc_fix_t, JPC_FIX_FRACBITS, x);
00124 }
00125
00126 JAS_ATTRIBUTE_CONST
00127 static inline double jpc_fixtodbl(jpc_fix_t x)
00128 {
00129     return JAS_FIXTODBL(jpc_fix_t, JPC_FIX_FRACBITS, x);
00130 }
00131
00132 JAS_ATTRIBUTE_CONST
00133 static inline jpc_fix_t jpc_dbltofix(double x)
00134 {
00135     return JAS_DBLTOFIX(jpc_fix_t, JPC_FIX_FRACBITS, x);
00136 }
00137
00138 JAS_ATTRIBUTE_CONST
00139 static inline jpc_fix_t jpc_fix_add(jpc_fix_t x, jpc_fix_t y)
00140 {
00141     return JAS_FIX_ADD(jpc_fix_t, JPC_FIX_FRACBITS, x, y);
00142 }
00143
00144 JAS_ATTRIBUTE_CONST
00145 static inline jpc_fix_t jpc_fix_sub(jpc_fix_t x, jpc_fix_t y)
00146 {
00147     return JAS_FIX_SUB(jpc_fix_t, JPC_FIX_FRACBITS, x, y);
00148 }
00149
00150 JAS_ATTRIBUTE_CONST
00151 static inline jpc_fix_t jpc_fix_mul(jpc_fix_big_t x, jpc_fix_big_t y)
00152 {
00153     return JAS_FIX_MUL(jpc_fix_t, JPC_FIX_FRACBITS, jpc_fix_big_t, x, y);
00154 }
00155
00156 JAS_ATTRIBUTE_CONST
00157 static inline jpc_fix_big_t jpc_fix_mulbyint(jpc_fix_big_t x, int y)
00158 {
00159     return JAS_FIX_MUL(jpc_fix_t, JPC_FIX_FRACBITS, jpc_fix_big_t, x, y);
00160 }
00161
00162 JAS_ATTRIBUTE_CONST
00163 static inline jpc_fix_t jpc_fix_div(jpc_fix_big_t x, jpc_fix_t y)
00164 {
00165     return JAS_FIX_DIV(jpc_fix_t, JPC_FIX_FRACBITS, jpc_fix_big_t, x, y);
00166 }
00167
00168 JAS_ATTRIBUTE_CONST
00169 static inline jpc_fix_t jpc_fix_neg(jpc_fix_t x)
00170 {
00171     return JAS_FIX_NEG(jpc_fix_t, JPC_FIX_FRACBITS, x);
00172 }
00173
00174 // #define      jpc_fix_asl(x, n)      JAS_FIX_ASL(jpc_fix_t, JPC_FIX_FRACBITS, x, n)
00175 // #define      jpc_fix_asr(x, n)      JAS_FIX_ASR(jpc_fix_t, JPC_FIX_FRACBITS, x, n)
00176
00177 #ifndef JAS_ENABLE_32BIT
00178 #define jpc_fix_asl jas_least32_asl
00179 #define jpc_fix_asr jas_least32_asr
00180 #else
00181 #define jpc_fix_asl jas_fast32_asl
00182 #define jpc_fix_asr jas_fast32_asr
00183 #endif
00184
00185 #define jpc_fix_pluseq(x, y)      JAS_FIX_PLUSEQ(jpc_fix_t, JPC_FIX_FRACBITS, x, y)
00186 #define jpc_fix_minuseq(x, y)    JAS_FIX_MINUSEQ(jpc_fix_t, JPC_FIX_FRACBITS, x, y)

```

```

00187 #define jpc_fix_muleq(x, y) \
00188     JAS_FIX_MULEQ(jpc_fix_t, JPC_FIX_FRACBITS, jpc_fix_big_t, x, y)
00189
00190 JAS_ATTRIBUTE_CONST
00191 static inline jpc_fix_t jpc_fix_abs(jpc_fix_t x)
00192 {
00193     return JAS_FIX_ABS(jpc_fix_t, JPC_FIX_FRACBITS, x);
00194 }
00195
00196 JAS_ATTRIBUTE_CONST
00197 static inline bool jpc_fix_isint(jpc_fix_t x)
00198 {
00199     return JAS_FIX_ISINT(jpc_fix_t, JPC_FIX_FRACBITS, x);
00200 }
00201
00202 JAS_ATTRIBUTE_CONST
00203 static inline int jpc_fix_sgn(jpc_fix_t x)
00204 {
00205     return JAS_FIX_SGN(jpc_fix_t, JPC_FIX_FRACBITS, x);
00206 }
00207
00208 JAS_ATTRIBUTE_CONST
00209 static inline jpc_fix_t jpc_fix_round(jpc_fix_t x)
00210 {
00211     return JAS_FIX_ROUND(jpc_fix_t, JPC_FIX_FRACBITS, x);
00212 }
00213
00214 JAS_ATTRIBUTE_CONST
00215 static inline jpc_fix_t jpc_fix_floor(jpc_fix_t x)
00216 {
00217     return JAS_FIX_FLOOR(jpc_fix_t, JPC_FIX_FRACBITS, x);
00218 }
00219
00220 /*****
00221  * Extra macros for convenience.
00222  *****/
00223
00224 /* Compute the sum of three fixed-point numbers. */
00225 JAS_ATTRIBUTE_CONST
00226 static inline jpc_fix_t jpc_fix_add3(jpc_fix_t x, jpc_fix_t y, jpc_fix_t z)
00227 {
00228     return jpc_fix_add(jpc_fix_add(x, y), z);
00229 }
00230
00231 #endif

```

## 11.49 jpcflt.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *

```

```

00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065  * Floating-Point Class
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_FLT_H
00071 #define JPC_FLT_H
00072
00073 /* The code ought to be modified so this type is not used at all. */
00074 /* Very few places in the code rely on floating-point arithmetic, aside
00075  * from conversions in printf's. */
00076 typedef double jpcflt_t;
00077
00078 #endif

```

## 11.50 jpc\_math.h

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or

```



```

00026 * substantial portions of the Software.
00027 *
00028 * 2. The name of a copyright holder shall not be used to endorse or
00029 * promote products derived from the Software without specific prior
00030 * written permission.
00031 *
00032 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037 * PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 #ifndef JPC_MATH_H
00063 #define JPC_MATH_H
00064
00065 /*****
00066 * Includes
00067 *****/
00068
00069 #include "jasper/jas_config.h"
00070
00071 #include "jpc_fix.h"
00072
00073 /*****
00074 * Macros
00075 *****/
00076
00077 /* Compute the floor of the quotient of two integers. */
00078 #define JPC_FLOORDIV(x, y) ((x) / (y))
00079
00080 /* Compute the ceiling of the quotient of two integers. */
00081 #define JPC_CEILDIV(x, y) (((x) + (y) - 1) / (y))
00082
00083 /* Compute the floor of (x / 2^y). */
00084 #define JPC_FLOORDIVPOW2(x, y) ((x) >> (y))
00085
00086 /* Compute the ceiling of (x / 2^y). */
00087 #define JPC_CEILDIVPOW2(x, y) (((x) + (1 << (y)) - 1) >> (y))
00088
00089 /*****
00090 * Functions.
00091 *****/
00092
00093 /* Calculate the bit position of the first leading one in a nonnegative
00094 integer. */
00095 JAS_ATTRIBUTE_CONST
00096 int jpc_int_firstone(int x);
00097
00098 JAS_ATTRIBUTE_CONST
00099 int jpc_fix_firstone(jpc_fix_t x);
00100
00101 /* Calculate the integer quantity floor(log2(x)), where x is a positive
00102 integer. */
00103 JAS_ATTRIBUTE_CONST
00104 unsigned jpc_floorlog2(uint_fast32_t x);
00105
00106 #endif

```

## 11.51 jpc\_mct.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Multicomponent Transform Code
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_MCT_H
00071 #define JPC_MCT_H
00072
00073 /*****\
00074  * Includes.
00075  */
00076 /*****\
00077 #include "jpc_fix.h"
00078

```

```

00079 #include "jasper/jas_seq.h"
00080
00081 /*****
00082  * Constants.
00083  *****/
00084
00085 /*
00086  * Multicomponent transform IDs.
00087  */
00088
00089 #define JPC_MCT_NONE    0
00090 #define JPC_MCT_ICT     1
00091 #define JPC_MCT_RCT     2
00092
00093 /*****
00094  * Functions.
00095  *****/
00096
00097 /* Calculate the forward RCT. */
00098 void jpc_rct(jas_matrix_t *c0, jas_matrix_t *c1, jas_matrix_t *c2);
00099
00100 /* Calculate the inverse RCT. */
00101 void jpc_ircrct(jas_matrix_t *c0, jas_matrix_t *c1, jas_matrix_t *c2);
00102
00103 /* Calculate the forward ICT. */
00104 void jpc_ict(jas_matrix_t *c0, jas_matrix_t *c1, jas_matrix_t *c2);
00105
00106 /* Calculate the inverse ICT. */
00107 void jpc_iict(jas_matrix_t *c0, jas_matrix_t *c1, jas_matrix_t *c2);
00108
00109 #endif

```

## 11.52 jpc\_mqcod.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION

```

```

00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064  /*
00065  * MQ Arithmetic Coder
00066  *
00067  * $Id$
00068  */
00069
00070  #ifndef JPC_MQCOD_H
00071  #define JPC_MQCOD_H
00072
00073  /*****\
00074  * Includes.
00075  \*****/
00076
00077  #include "jasper/jas_types.h"
00078
00079  /*****\
00080  * Types.
00081  \*****/
00082
00083  /*
00084  * MQ coder context information.
00085  */
00086
00087  typedef struct {
00088
00089      /* The most probable symbol (MPS). */
00090      bool mps;
00091
00092      /* The state index. */
00093      int_least8_t ind;
00094
00095  } jpc_mqctx_t;
00096
00097  /*
00098  * MQ coder state table entry.
00099  */
00100
00101  typedef struct jpc_mqstate_s {
00102
00103      /* The Qe value. */
00104      uint_least16_t qeval;
00105
00106      /* The MPS. */
00107      bool mps;
00108
00109      /* The NMPS state. */
00110      const struct jpc_mqstate_s *nmpps;
00111
00112      /* The NLPS state. */
00113      const struct jpc_mqstate_s *nlps;
00114
00115  } jpc_mqstate_t;
00116
00117  /*****\
00118  * Data.
00119  \*****/
00120
00121  /* The state table for the MQ coder. */
00122  extern const jpc_mqstate_t jpc_mqstates[];
00123
00124  #endif

```

## 11.53 jpc\_mqdec.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2003 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * MQ Arithmetic Decoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_MQDEC_H
00071 #define JPC_MQDEC_H
00072
00073 /*****\
00074  * Includes.
00075  * *****/
00076
00077 #include "jasper/jas_types.h"
00078 #include "jasper/jas_stream.h"

```

```

00079
00080 #include "jpc_mqcod.h"
00081
00082 #include <stdio.h>
00083
00084 /*****
00085  * Types.
00086  *****/
00087
00088 /* MQ arithmetic decoder. */
00089
00090 typedef struct {
00091     /* The C register. */
00092     uint_least32_t creg;
00093
00094     /* The A register. */
00095     uint_least32_t areg;
00096
00097     /* The CT register. */
00098     uint_least32_t ctreg;
00099
00100     /* The current context. */
00101     const jpc_mqstate_t **curctx;
00102
00103     /* The per-context information. */
00104     const jpc_mqstate_t **ctxs;
00105
00106     /* The maximum number of contexts. */
00107     unsigned maxctxs;
00108
00109     /* The stream from which to read data. */
00110     jas_stream_t *in;
00111
00112     /* The last character read. */
00113     jas_uchar inbuffer;
00114
00115     /* The EOF indicator. */
00116     bool eof;
00117 } jpc_mqdec_t;
00118
00119 } jpc_mqdec_t;
00120
00121 /*****
00122  * Functions/macros for construction and destruction.
00123  *****/
00124
00125 /* Create a MQ decoder. */
00126 jpc_mqdec_t *jpc_mqdec_create(unsigned maxctxs, jas_stream_t *in);
00127
00128 /* Destroy a MQ decoder. */
00129 void jpc_mqdec_destroy(jpc_mqdec_t *dec);
00130
00131 /*****
00132  * Functions/macros for initialization.
00133  *****/
00134
00135 /* Set the input stream associated with a MQ decoder. */
00136 void jpc_mqdec_setinput(jpc_mqdec_t *dec, jas_stream_t *in);
00137
00138 /* Initialize a MQ decoder. */
00139 void jpc_mqdec_init(jpc_mqdec_t *dec);
00140
00141 /*****
00142  * Functions/macros for manipulating contexts.
00143  *****/
00144
00145 /* Set the current context for a MQ decoder. */
00146 static inline void jpc_mqdec_setcurctx(jpc_mqdec_t *dec, unsigned ctxno)
00147 {
00148     dec->curctx = &dec->ctxs[ctxno];
00149 }
00150
00151 /* Set the state information for all contexts of a MQ decoder. */
00152 void jpc_mqdec_setctxs(const jpc_mqdec_t *dec, unsigned numctxs, const jpc_mqctx_t *ctxs);
00153
00154 /*****
00155  * Functions/macros for decoding bits.
00156  *****/
00157
00158 /* Decode a symbol. */
00159 #ifndef NDEBUG

```

```

00160 #define jpc_mqdec_getbit(dec) \
00161     jpc_mqdec_getbit_macro(dec)
00162 #else
00163 #define jpc_mqdec_getbit(dec) \
00164     jpc_mqdec_getbit_func(dec)
00165 #endif
00166
00167 /* Decode a symbol (assuming an unskewed probability distribution). */
00168 #ifdef NDEBUB
00169 #define jpc_mqdec_getbitnoskew(dec) \
00170     jpc_mqdec_getbit_macro(dec)
00171 #else
00172 #define jpc_mqdec_getbitnoskew(dec) \
00173     jpc_mqdec_getbit_func(dec)
00174 #endif
00175
00176 /*****
00177  * Functions/macros for debugging.
00178  *****/
00179
00180 /* Dump the MQ decoder state for debugging. */
00181 void jpc_mqdec_dump(const jpc_mqdec_t *dec, FILE *out);
00182
00183 /*****
00184  * EVERYTHING BELOW THIS POINT IS IMPLEMENTATION SPECIFIC AND NOT PART OF THE
00185  * APPLICATION INTERFACE. DO NOT RELY ON ANY OF THE INTERNAL FUNCTIONS/MACROS
00186  * GIVEN BELOW.
00187  *****/
00188
00189 bool jpc_mqdec_mpsexchrenormd(jpc_mqdec_t *dec);
00190 bool jpc_mqdec_lpsexchrenormd(jpc_mqdec_t *dec);
00191
00192 JAS_FORCE_INLINE
00193 static bool jpc_mqdec_getbit_macro(jpc_mqdec_t *dec)
00194 {
00195     const jpc_mqstate_t *const state = *dec->curctx;
00196     dec->areg -= state->qeval;
00197
00198     if (dec->creg >= (uint_least32_t)state->qeval << 16) {
00199         dec->creg -= (uint_least32_t)state->qeval << 16;
00200         return dec->areg & 0x8000
00201             ? state->mps
00202             : jpc_mqdec_mpsexchrenormd(dec);
00203     } else {
00204         return jpc_mqdec_lpsexchrenormd(dec);
00205     }
00206 }
00207
00208 JAS_FORCE_INLINE
00209 static bool jpc_mqdec_mpsexchange(uint_least32_t areg, uint_least32_t delta, const jpc_mqstate_t **curctx)
00210 {
00211     if (areg < delta) {
00212         const jpc_mqstate_t *state = *curctx;
00213         /* LPS decoded. */
00214         *curctx = state->nmps;
00215         return !state->mps;
00216     } else {
00217         const jpc_mqstate_t *state = *curctx;
00218         /* MPS decoded. */
00219         *curctx = state->nmps;
00220         return state->mps;
00221     }
00222 }
00223
00224 JAS_FORCE_INLINE
00225 static bool jpc_mqdec_lpsexchange(uint_least32_t *areg_p, uint_least32_t delta, const jpc_mqstate_t
00226 **curctx)
00227 {
00228     if (*areg_p >= delta) {
00229         const jpc_mqstate_t *state = *curctx;
00230         *areg_p = delta;
00231         *curctx = state->nmps;
00232         return !state->mps;
00233     } else {
00234         const jpc_mqstate_t *state = *curctx;
00235         *areg_p = delta;
00236         *curctx = state->nmps;
00237         return state->mps;
00238     }
00239 }

```

```

00240
00241 bool jpc_mqdec_getbit_func(jpc_mqdec_t *dec);
00242
00243 #endif

```

## 11.54 jpc\_mqenc.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * MQ Arithmetic Encoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_MQENC_H

```



```

00071 #define JPC_MQENC_H
00072
00073 /*****
00074  * Includes.
00075  *****/
00076
00077 #include "jasper/jas_types.h"
00078 #include "jasper/jas_stream.h"
00079
00080 #include "jpc_mqcod.h"
00081
00082 #include <stdio.h>
00083
00084 /*****
00085  * Constants.
00086  *****/
00087
00088 /*
00089  * Termination modes.
00090  */
00091
00092 #define JPC_MQENC_DEFTERM      0      /* default termination */
00093 #define JPC_MQENC_PTERM       1      /* predictable termination */
00094
00095 /*****
00096  * Types.
00097  *****/
00098
00099 /* MQ arithmetic encoder class. */
00100
00101 typedef struct {
00102
00103     /* The C register. */
00104     uint_least32_t creg;
00105
00106     /* The A register. */
00107     uint_least32_t areg;
00108
00109     /* The CT register. */
00110     uint_least32_t ctreg;
00111
00112     /* The maximum number of contexts. */
00113     unsigned maxctxs;
00114
00115     /* The per-context information. */
00116     const jpc_mqstate_t **ctxs;
00117
00118     /* The current context. */
00119     const jpc_mqstate_t **curctx;
00120
00121     /* The stream for encoder output. */
00122     jas_stream_t *out;
00123
00124     /* The byte buffer (i.e., the B variable in the standard). */
00125     int_least16_t outbuf;
00126
00127     /* The last byte output. */
00128     int_least16_t lastbyte;
00129
00130     /* The error indicator. */
00131     bool err;
00132 } jpc_mqenc_t;
00133
00134 /* MQ arithmetic encoder state information. */
00135
00136 typedef struct {
00137
00138     /* The A register. */
00139     unsigned areg;
00140
00141     /* The C register. */
00142     unsigned creg;
00143
00144     /* The CT register. */
00145     unsigned ctreg;
00146
00147     /* The last byte output by the encoder. */
00148     int lastbyte;
00149 } jpc_mqencstate_t;
00150
00151 } jpc_mqencstate_t;

```

```

00152
00153 /*****
00154  * Functions/macros for construction and destruction.
00155  *****/
00156
00157 /* Create a MQ encoder. */
00158 jpc_mqenc_t *jpc_mqenc_create(unsigned maxctxs, jas_stream_t *out);
00159
00160 /* Destroy a MQ encoder. */
00161 void jpc_mqenc_destroy(jpc_mqenc_t *enc);
00162
00163 /*****
00164  * Functions/macros for initialization.
00165  *****/
00166
00167 /* Initialize a MQ encoder. */
00168 void jpc_mqenc_init(jpc_mqenc_t *enc);
00169
00170 /*****
00171  * Functions/macros for context manipulation.
00172  *****/
00173
00174 /* Set the current context. */
00175 static inline void jpc_mqenc_setcurctx(jpc_mqenc_t *enc, unsigned ctxno) {
00176     enc->curctx = &enc->ctxs[ctxno];
00177 }
00178
00179 /* Set the state information for multiple contexts. */
00180 void jpc_mqenc_setctxs(jpc_mqenc_t *enc, unsigned numctxs, const jpc_mqctx_t *ctxs);
00181
00182 /*****
00183  * Miscellaneous functions/macros.
00184  *****/
00185
00186 /* Get the error state of a MQ encoder. */
00187 static inline bool jpc_mqenc_error(const jpc_mqenc_t *enc) {
00188     return enc->err;
00189 }
00190
00191 /* Get the current encoder state. */
00192 void jpc_mqenc_getstate(const jpc_mqenc_t *enc, jpc_mqencstate_t *state);
00193
00194 /* Terminate the code. */
00195 int jpc_mqenc_flush(jpc_mqenc_t *enc, int termmode);
00196
00197 /*****
00198  * Functions/macros for encoding bits.
00199  *****/
00200
00201 /*****
00202  * Functions/macros for debugging.
00203  *****/
00204
00205 int jpc_mqenc_dump(const jpc_mqenc_t *mqenc, FILE *out);
00206
00207 /*****
00208  * Implementation-specific details.
00209  *****/
00210
00211 /* Note: These function prototypes are included only to satisfy the
00212  needs of the mqenc_putbit_macro macro. Do not call any of these
00213  functions directly. */
00214 int jpc_mqenc_codemps2(jpc_mqenc_t *enc);
00215 int jpc_mqenc_codelps(jpc_mqenc_t *enc);
00216
00217 int jpc_mqenc_putbit(jpc_mqenc_t *enc, bool bit);
00218
00219 #endif

```

## 11.55 jpc\_qmfb.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2004 Michael David Adams.
00005  * All rights reserved.
00006  */

```

```

00007
00008 /* __START_OF_JASPER_LICENSE__
00009 *
00010 * Jasper License Version 2.0
00011 *
00012 * Copyright (c) 2001-2006 Michael David Adams
00013 * Copyright (c) 1999-2000 Image Power, Inc.
00014 * Copyright (c) 1999-2000 The University of British Columbia
00015 *
00016 * All rights reserved.
00017 *
00018 * Permission is hereby granted, free of charge, to any person (the
00019 * "User") obtaining a copy of this software and associated documentation
00020 * files (the "Software"), to deal in the Software without restriction,
00021 * including without limitation the rights to use, copy, modify, merge,
00022 * publish, distribute, and/or sell copies of the Software, and to permit
00023 * persons to whom the Software is furnished to do so, subject to the
00024 * following conditions:
00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065 * Quadrature Mirror-Image Filter Bank (QMFB) Routines
00066 *
00067 * $Id$
00068 */
00069
00070 #ifndef JPC_QMFB_H
00071 #define JPC_QMFB_H
00072
00073 /*****\
00074 * Includes.
00075 \*****/
00076
00077 #include "jpc_fix.h"
00078
00079 /*****\
00080 * Constants.
00081 \*****/
00082
00083 /* QMFB IDs. */
00084 #define JPC_QMFB1D_FT 1 /* 5/3 */
00085 #define JPC_QMFB1D_NS 2 /* 9/7 */
00086
00087 /*****\

```

```

00088 * Types.
00089 \*****/
00090
00091 /*****\
00092 * Functions.
00093 \*****/
00094
00095 #if !defined(JPC_QMFB_COLGRPSIZE)
00096 /* The number of columns to group together during the vertical processing
00097 stage of the wavelet transform. */
00098 /* The default value for this parameter is probably not optimal for
00099 any particular platform. Hopefully, it is not too unreasonable, however. */
00100 #define JPC_QMFB_COLGRPSIZE 16
00101 #endif
00102
00103 typedef struct {
00104     int (*analyze)(jpc_fix_t *, int, int, int, int, int);
00105     int (*synthesize)(jpc_fix_t *, int, int, int, int, int);
00106     const double *lpenergywts;
00107     const double *hpenergywts;
00108 } jpc_qmfb2d_t;
00109
00110 extern const jpc_qmfb2d_t jpc_ft_qmfb2d;
00111 extern const jpc_qmfb2d_t jpc_ns_qmfb2d;
00112
00113 #endif

```

## 11.56 jpc\_t1cod.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL

```

```

00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * $Id$
00066  */
00067
00068 #ifndef JPC_T1COD_H
00069 #define JPC_T1COD_H
00070
00071 /*****\
00072  * Includes.
00073  \*****/
00074
00075 #include "jpc_fix.h"
00076 #include "jpc_mqcod.h"
00077 #include "jpc_tsfb.h"
00078 #include "jasper/jas_math.h"
00079
00080 /*****\
00081  * Constants.
00082  \*****/
00083
00084 /* The number of bits used to index into various lookup tables. */
00085 #define JPC_NMSEDEC_BITS 7
00086 #define JPC_NMSEDEC_FRACBITS (JPC_NMSEDEC_BITS - 1)
00087
00088 /*
00089  * Segment types.
00090  */
00091
00092 enum jpc_segtype {
00093     JPC_SEG_INVALID,
00094
00095     /* MQ. */
00096     JPC_SEG_MQ,
00097
00098     /* Raw. */
00099     JPC_SEG_RAW,
00100 };
00101
00102 /* The nominal word size. */
00103 #define JPC_PREC 32
00104
00105 /* Tier-1 coding pass types. */
00106 enum jpc_passtype {
00107     JPC_SIGPASS, /*< significance */
00108     JPC_REFPASS, /*< refinement */
00109     JPC_CLNPASS, /*< cleanup */
00110 };
00111
00112 /*
00113  * Per-sample state information for tier-1 coding.
00114  */
00115
00116 /* The northeast neighbour has been found to be significant. */
00117 #define JPC_NESIG 0x0001
00118 /* The southeast neighbour has been found to be significant. */
00119 #define JPC_SESIG 0x0002
00120 /* The southwest neighbour has been found to be significant. */
00121 #define JPC_SWSIG 0x0004
00122 /* The northwest neighbour has been found to be significant. */
00123 #define JPC_NWSIG 0x0008
00124 /* The north neighbour has been found to be significant. */
00125 #define JPC_NSIG 0x0010
00126 /* The east neighbour has been found to be significant. */
00127 #define JPC_ESIG 0x0020
00128 /* The south neighbour has been found to be significant. */
00129 #define JPC_SSIG 0x0040
00130

```

```

00131 /* The west neighbour has been found to be significant. */
00132 #define JPC_WSIG      0x0080
00133 /* The significance mask for 8-connected neighbours. */
00134 #define JPC_OTH SIGMSK \
00135     (JPC_NSIG | JPC_NESIG | JPC_ESIG | JPC_SESIG | JPC_SSIG | JPC_SWSIG | JPC_WSIG | JPC_NWSIG)
00136 /* The significance mask for 4-connected neighbours. */
00137 #define JPC_PRIMSIGMSK (JPC_NSIG | JPC_ESIG | JPC_SSIG | JPC_WSIG)
00138
00139 /* The north neighbour is negative in value. */
00140 #define JPC_NSGN      0x0100
00141 /* The east neighbour is negative in value. */
00142 #define JPC_ESGN      0x0200
00143 /* The south neighbour is negative in value. */
00144 #define JPC_SSGN      0x0400
00145 /* The west neighbour is negative in value. */
00146 #define JPC_WSGN      0x0800
00147 /* The sign mask for 4-connected neighbours. */
00148 #define JPC_SGNMSK     (JPC_NSGN | JPC_ESGN | JPC_SSGN | JPC_WSGN)
00149
00150 /* This sample has been found to be significant. */
00151 #define JPC_SIG        0x1000
00152 /* The sample has been refined. */
00153 #define JPC_REFINE     0x2000
00154 /* This sample has been processed during the significance pass. */
00155 #define JPC_VISIT      0x4000
00156
00157 /* The number of aggregation contexts. */
00158 #define JPC_NUMAGGCTXS 1
00159 /* The number of zero coding contexts. */
00160 #define JPC_NUMZCCTXS  9
00161 /* The number of magnitude contexts. */
00162 #define JPC_NUMMAGCTXS 3
00163 /* The number of sign coding contexts. */
00164 #define JPC_NUMSCCTXS  5
00165 /* The number of uniform contexts. */
00166 #define JPC_NUMUCTXS    1
00167
00168 /* The context ID for the first aggregation context. */
00169 #define JPC_AGGCTXNO   0
00170 /* The context ID for the first zero coding context. */
00171 #define JPC_ZCCTXNO    (JPC_AGGCTXNO + JPC_NUMAGGCTXS)
00172 /* The context ID for the first magnitude context. */
00173 #define JPC_MAGCTXNO   (JPC_ZCCTXNO + JPC_NUMZCCTXS)
00174 /* The context ID for the first sign coding context. */
00175 #define JPC_SCCTXNO    (JPC_MAGCTXNO + JPC_NUMMAGCTXS)
00176 /* The context ID for the first uniform context. */
00177 #define JPC_UCTXNO     (JPC_SCCTXNO + JPC_NUMSCCTXS)
00178 /* The total number of contexts. */
00179 #define JPC_NUMCTXS    (JPC_UCTXNO + JPC_NUMUCTXS)
00180
00181 /*****
00182  * External data.
00183  \*****/
00184
00185 /* These lookup tables are used by various macros/functions. */
00186 /* Do not access these lookup tables directly. */
00187 extern uint_least8_t jpc_zcctxnolut[];
00188 extern bool jpc_spblut[];
00189 extern uint_least8_t jpc_scctxnolut[];
00190 extern uint_least8_t jpc_magctxnolut[];
00191 extern jpc_fix_t jpc_refnmsedec[];
00192 extern jpc_fix_t jpc_signmsedec[];
00193 extern jpc_fix_t jpc_refnmsedec0[];
00194 extern jpc_fix_t jpc_signmsedec0[];
00195
00196 /* The initial settings for the MQ contexts. */
00197 extern jpc_mqctx_t jpc_mqctxs[];
00198
00199 /*****
00200  * Functions and macros.
00201  \*****/
00202
00203 /* Arithmetic shift right (with ability to shift left also). */
00204 JAS_ATTRIBUTE_CONST
00205 static inline jpc_fix_t JPC_ASR(jpc_fix_t x, int n)
00206 {
00207     return n >= 0
00208         ? x » n
00209         : x « -n;
00210 }
00211

```

```

00212 /* Get the zero coding context. */
00213 JAS_ATTRIBUTE_CONST
00214 static inline uint_least8_t JPC_GETZCCTXNO(unsigned f, enum jpc_tsfb_orient orient)
00215 {
00216     return jpc_zcctxnolut[((unsigned)orient << 8) | (f & JPC_OTHSIGMSK)];
00217 }
00218
00219 /* Get the sign prediction bit. */
00220 JAS_ATTRIBUTE_CONST
00221 static inline bool JPC_GETSPB(unsigned f)
00222 {
00223     return jpc_spblut[(f & (JPC_PRIMSIGMSK | JPC_SGNMSK)) >> 4];
00224 }
00225
00226 /* Get the sign coding context. */
00227 JAS_ATTRIBUTE_CONST
00228 static inline uint_least8_t JPC_GETSCCTXNO(unsigned f)
00229 {
00230     return jpc_scctxnolut[(f & (JPC_PRIMSIGMSK | JPC_SGNMSK)) >> 4];
00231 }
00232
00233 /* Get the magnitude context. */
00234 JAS_ATTRIBUTE_CONST
00235 static inline uint_least8_t JPC_GETMAGCTXNO(unsigned f)
00236 {
00237     return jpc_magctxnolut[(f & JPC_OTHSIGMSK) | (((f & JPC_REFINE) != 0) << 11)];
00238 }
00239
00240 /* Get the normalized MSE reduction for significance passes. */
00241 JAS_ATTRIBUTE_CONST
00242 static inline jpc_fix_t JPC_GETSIGNMSEDEC(jpc_fix_t x, int bitpos)
00243 {
00244     return bitpos > JPC_NMSEDEC_FRACBITS
00245         ? jpc_signmsedec[JPC_ASR(x, bitpos - JPC_NMSEDEC_FRACBITS) & JAS_ONES(JPC_NMSEDEC_BITS)]
00246         : jpc_signmsedec0[JPC_ASR(x, bitpos - JPC_NMSEDEC_FRACBITS) & JAS_ONES(JPC_NMSEDEC_BITS)];
00247 }
00248
00249 /* Get the normalized MSE reduction for refinement passes. */
00250 JAS_ATTRIBUTE_CONST
00251 static inline jpc_fix_t JPC_GETREFNMSEDEC(jpc_fix_t x, int bitpos)
00252 {
00253     return bitpos > JPC_NMSEDEC_FRACBITS
00254         ? jpc_refnmsedec[JPC_ASR(x, bitpos - JPC_NMSEDEC_FRACBITS) & JAS_ONES(JPC_NMSEDEC_BITS)]
00255         : jpc_refnmsedec0[JPC_ASR(x, bitpos - JPC_NMSEDEC_FRACBITS) & JAS_ONES(JPC_NMSEDEC_BITS)];
00256 }
00257
00258 /* Update the per-sample state information. */
00259 static inline void JPC_UPDATEFLAGS4(jpc_fix_t *fp, unsigned rowstep, bool s, bool vcausalflag)
00260 {
00261     jpc_fix_t *np = fp - rowstep;
00262     jpc_fix_t *sp = fp + rowstep;
00263     if (vcausalflag) {
00264         sp[-1] |= JPC_NESIG;
00265         sp[1] |= JPC_NWSIG;
00266         if (s) {
00267             *sp |= JPC_NSIG | JPC_NSGN;
00268             fp[-1] |= JPC_ESIG | JPC_ESGN;
00269             fp[1] |= JPC_WSIG | JPC_WSGN;
00270         } else {
00271             *sp |= JPC_NSIG;
00272             fp[-1] |= JPC_ESIG;
00273             fp[1] |= JPC_WSIG;
00274         }
00275     } else {
00276         np[-1] |= JPC_SESIG;
00277         np[1] |= JPC_SWSIG;
00278         sp[-1] |= JPC_NESIG;
00279         sp[1] |= JPC_NWSIG;
00280         if (s) {
00281             *np |= JPC_SSIG | JPC_SSGN;
00282             *sp |= JPC_NSIG | JPC_NSGN;
00283             fp[-1] |= JPC_ESIG | JPC_ESGN;
00284             fp[1] |= JPC_WSIG | JPC_WSGN;
00285         } else {
00286             *np |= JPC_SSIG;
00287             *sp |= JPC_NSIG;
00288             fp[-1] |= JPC_ESIG;
00289             fp[1] |= JPC_WSIG;
00290         }
00291     }
00292 }

```

```

00293
00294 /* Initialize the lookup tables used by the codec. */
00295 void jpc_initluts(void);
00296
00297 /* Get the nominal gain associated with a particular band. */
00298 JAS_ATTRIBUTE_CONST
00299 unsigned JPC_NOMINALGAIN(unsigned qmfbid, unsigned numlvs, unsigned lvlno, enum jpc_tsfb_orient orient);
00300
00301 /* Get the coding pass type. */
00302 JAS_ATTRIBUTE_CONST
00303 enum jpc_passtype JPC_PASSTYPE(unsigned passno);
00304
00305 /* Get the segment type. */
00306 JAS_ATTRIBUTE_CONST
00307 enum jpc_segtype JPC_SEGTYPE(unsigned passno, unsigned firstpassno, bool bypass);
00308
00309 /* Get the number of coding passes in the segment. */
00310 JAS_ATTRIBUTE_CONST
00311 unsigned JPC_SEGPASSCNT(unsigned passno, unsigned firstpassno, unsigned numpasses, bool bypass,
00312     bool termall);
00313
00314 /* Is the coding pass terminated? */
00315 JAS_ATTRIBUTE_CONST
00316 bool JPC_ISTERMINATED(unsigned passno, unsigned firstpassno, unsigned numpasses, bool termall,
00317     bool lazy);
00318
00319 #endif

```

## 11.57 jpc\_t1dec.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS

```



```

00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065 * Tier 1 Decoder
00066 *
00067 * $Id$
00068 */
00069
00070 #ifndef JPC_T1DEC_H
00071 #define JPC_T1DEC_H
00072
00073 /*****
00074 * Includes.
00075 *****/
00076
00077 #include "jpc_dec.h"
00078
00079 /*****
00080 * Functions.
00081 *****/
00082
00083 /* Decode all of the code blocks for a particular tile. */
00084 int jpc_dec_decodecbks(jpc_dec_t *dec, jpc_dec_tile_t *tile);
00085
00086 #endif

```

## 11.58 jpc\_t1enc.h

```

00001 /*
00002 * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003 * British Columbia.
00004 * Copyright (c) 2001-2002 Michael David Adams.
00005 * All rights reserved.
00006 */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009 *
00010 * JasPer License Version 2.0
00011 *
00012 * Copyright (c) 2001-2006 Michael David Adams
00013 * Copyright (c) 1999-2000 Image Power, Inc.
00014 * Copyright (c) 1999-2000 The University of British Columbia
00015 *
00016 * All rights reserved.
00017 *
00018 * Permission is hereby granted, free of charge, to any person (the
00019 * "User") obtaining a copy of this software and associated documentation
00020 * files (the "Software"), to deal in the Software without restriction,
00021 * including without limitation the rights to use, copy, modify, merge,
00022 * publish, distribute, and/or sell copies of the Software, and to permit
00023 * persons to whom the Software is furnished to do so, subject to the
00024 * following conditions:
00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER

```

```

00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065  * Tier 1 Encoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_T1ENC_H
00071 #define JPC_T1ENC_H
00072
00073 /*****
00074  * Includes.
00075  *****/
00076
00077 #include "jpc_enc.h"
00078
00079 /*****
00080  * Functions.
00081  *****/
00082
00083 /* Encode all of the code blocks. */
00084 int jpc_enc_encblk(jpc_enc_t *enc);
00085
00086 #endif

```

## 11.59 jpc\_t2cod.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  * British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the

```

```

00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064  /*
00065  * Tier-2 Coding Library
00066  *
00067  * $Id$
00068  */
00069
00070  #ifndef JPC_T2COD_H
00071  #define JPC_T2COD_H
00072
00073  /*****
00074  * Includes.
00075  *****/
00076
00077  #include "jpc_cs.h"
00078
00079  /*****
00080  * Types.
00081  *****/
00082
00083  /* Progression change list. */
00084
00085  typedef struct {
00086
00087      /* The number of progression changes. */
00088      unsigned numpchgs;
00089
00090      /* The maximum number of progression changes that can be accomodated
00091       without growing the progression change array. */
00092      unsigned maxpchgs;
00093
00094      /* The progression changes. */
00095      jpc_pchg_t **pchgs;
00096
00097  } jpc_pchglist_t;
00098
00099  /* Packet iterator per-resolution-level information. */
00100
00101  typedef struct {
00102
00103      /* The number of precincts. */
00104      unsigned numprcs;

```

```
00105
00106     /* The last layer processed for each precinct. */
00107     unsigned *prclyrnos;
00108
00109     /* The precinct width exponent. */
00110     unsigned prcwidthexpn;
00111
00112     /* The precinct height exponent. */
00113     unsigned prcheightexpn;
00114
00115     /* The number of precincts spanning the resolution level in the horizontal
00116        direction. */
00117     unsigned numhprcs;
00118
00119 } jpc_pirlvl_t;
00120
00121 /* Packet iterator per-component information. */
00122
00123 typedef struct {
00124
00125     /* The number of resolution levels. */
00126     unsigned numrlvls;
00127
00128     /* The per-resolution-level information. */
00129     jpc_pirlvl_t *pirlvls;
00130
00131     /* The horizontal sampling period. */
00132     uint_fast32_t hsamp;
00133
00134     /* The vertical sampling period. */
00135     uint_fast32_t vsamp;
00136
00137 } jpc_picomp_t;
00138
00139 /* Packet iterator class. */
00140
00141 typedef struct {
00142
00143     /* The number of layers. */
00144     unsigned numlyrs;
00145
00146     /* The number of resolution levels. */
00147     unsigned maxrlvls;
00148
00149     /* The number of components. */
00150     unsigned numcomps;
00151
00152     /* The per-component information. */
00153     jpc_picomp_t *picomps;
00154
00155     /* The current component. */
00156     jpc_picomp_t *picomp;
00157
00158     /* The current resolution level. */
00159     jpc_pirlvl_t *pirlvl;
00160
00161     /* The number of the current component. */
00162     unsigned compno;
00163
00164     /* The number of the current resolution level. */
00165     unsigned rlvln;
00166
00167     /* The number of the current precinct. */
00168     unsigned prcno;
00169
00170     /* The number of the current layer. */
00171     unsigned lyrno;
00172
00173     /* The x-coordinate of the current position. */
00174     uint_fast32_t x;
00175
00176     /* The y-coordinate of the current position. */
00177     uint_fast32_t y;
00178
00179     /* The horizontal step size. */
00180     uint_fast32_t xstep;
00181
00182     /* The vertical step size. */
00183     uint_fast32_t ystep;
00184
00185     /* The x-coordinate of the top-left corner of the tile on the reference
```

```

00186         grid. */
00187         uint_fast32_t xstart;
00188
00189         /* The y-coordinate of the top-left corner of the tile on the reference
00190         grid. */
00191         uint_fast32_t ystart;
00192
00193         /* The x-coordinate of the bottom-right corner of the tile on the
00194         reference grid (plus one). */
00195         uint_fast32_t xend;
00196
00197         /* The y-coordinate of the bottom-right corner of the tile on the
00198         reference grid (plus one). */
00199         uint_fast32_t yend;
00200
00201         /* The current progression change. */
00202         const jpc_pchg_t *pchg;
00203
00204         /* The progression change list. */
00205         jpc_pchglist_t *pchglist;
00206
00207         /* The progression to use in the absense of explicit specification. */
00208         jpc_pchg_t defaultpchg;
00209
00210         /* The current progression change number. */
00211         int pchgno;
00212
00213         /* Is this the first time in the current progression volume? */
00214         bool prgvolfirst;
00215
00216         /* Is the current iterator value valid? */
00217         bool valid;
00218
00219         /* The current packet number. */
00220         int pktno;
00221
00222     } jpc_pi_t;
00223
00224     /*****
00225     * Functions/macros for packet iterators.
00226     \*****/
00227
00228     /* Create a packet iterator. */
00229     jpc_pi_t *jpc_pi_create0(void);
00230
00231     /* Destroy a packet iterator. */
00232     void jpc_pi_destroy(jpc_pi_t *pi);
00233
00234     /* Add a progression change to a packet iterator. */
00235     int jpc_pi_addpchg(jpc_pi_t *pi, jpc_pocpchgt_t *pchg);
00236
00237     /* Prepare a packet iterator for iteration. */
00238     int jpc_pi_init(jpc_pi_t *pi);
00239
00240     /* Set the iterator to the first packet. */
00241     int jpc_pi_begin(jpc_pi_t *pi);
00242
00243     /* Proceed to the next packet in sequence. */
00244     int jpc_pi_next(jpc_pi_t *pi);
00245
00246     /* Get the index of the current packet. */
00247     #define jpc_pi_getind(pi) ((pi)->pktno)
00248
00249     /* Get the component number of the current packet. */
00250     #define jpc_pi_cmptno(pi) (assert(pi->valid), (pi)->compno)
00251
00252     /* Get the resolution level of the current packet. */
00253     #define jpc_pi_rlvlno(pi) (assert(pi->valid), (pi)->rlvlno)
00254
00255     /* Get the layer number of the current packet. */
00256     #define jpc_pi_lyrno(pi) (assert(pi->valid), (pi)->lyrno)
00257
00258     /* Get the precinct number of the current packet. */
00259     #define jpc_pi_prcno(pi) (assert(pi->valid), (pi)->prcno)
00260
00261     /* Get the progression order for the current packet. */
00262     #define jpc_pi_prg(pi) (assert(pi->valid), (pi)->pchg->prgord)
00263
00264     /*****
00265     * Functions/macros for progression change lists.
00266     \*****/

```

```

00267
00268 /* Create a progression change list. */
00269 jpc_pchglst_t *jpc_pchglst_create(void);
00270
00271 /* Destroy a progression change list. */
00272 void jpc_pchglst_destroy(jpc_pchglst_t *pchglst);
00273
00274 /* Insert a new element into a progression change list. */
00275 int jpc_pchglst_insert(jpc_pchglst_t *pchglst, int pchgno, jpc_pchg_t *pchg);
00276
00277 /* Remove an element from a progression change list. */
00278 jpc_pchg_t *jpc_pchglst_remove(jpc_pchglst_t *pchglst, unsigned pchgno);
00279
00280 /* Get an element from a progression change list. */
00281 JAS_ATTRIBUTE_PURE
00282 const jpc_pchg_t *jpc_pchglst_get(const jpc_pchglst_t *pchglst, unsigned pchgno);
00283
00284 /* Copy a progression change list. */
00285 jpc_pchglst_t *jpc_pchglst_copy(const jpc_pchglst_t *pchglst);
00286
00287 /* Get the number of elements in a progression change list. */
00288 JAS_ATTRIBUTE_PURE
00289 unsigned jpc_pchglst_numpchgs(const jpc_pchglst_t *pchglst);
00290
00291 /*****
00292  * Functions/macros for progression changes.
00293  *****/
00294
00295 /* Destroy a progression change. */
00296 void jpc_pchg_destroy(jpc_pchg_t *pchg);
00297
00298 /* Copy a progression change. */
00299 jpc_pchg_t *jpc_pchg_copy(const jpc_pchg_t *pchg);
00300
00301 #endif

```

## 11.60 jpc\_t2dec.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO

```

```

00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Tier 2 Decoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_T2DEC_H
00071 #define JPC_T2DEC_H
00072
00073 /*****
00074  * Includes.
00075  *****/
00076 #include "jasper/jas_stream.h"
00077
00078 #include "jpc_dec.h"
00079 #include "jpc_t2cod.h"
00080
00081 /*****
00082  * Functions.
00083  *****/
00084
00085 /* Decode the packets for a tile-part. */
00086 int jpc_dec_decodepkts(jpc_dec_t *dec, jas_stream_t *pkthdrstream,
00087     jas_stream_t *in);
00088
00089 /* Create a packet iterator for the decoder. */
00090 jpc_pi_t *jpc_dec_pi_create(jpc_dec_t *dec, jpc_dec_tile_t *tile);
00091
00092 #endif

```

## 11.61 jpc\_t2enc.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,

```

```

00021 * including without limitation the rights to use, copy, modify, merge,
00022 * publish, distribute, and/or sell copies of the Software, and to permit
00023 * persons to whom the Software is furnished to do so, subject to the
00024 * following conditions:
00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060 *
00061 * __END_OF_JASPER_LICENSE__
00062 */
00063
00064 /*
00065  * Tier 2 Encoder
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_T2ENC_H
00071 #define JPC_T2ENC_H
00072
00073 /*****
00074  * Includes.
00075  *****/
00076
00077 #include "jasper/jas_stream.h"
00078
00079 #include "jpc_enc.h"
00080 #include "jpc_t2cod.h"
00081
00082 /*****
00083  * Functions.
00084  *****/
00085
00086 /* Encode the packets for a tile. */
00087 int jpc_enc_encpkts(jpc_enc_t *enc, jas_stream_t *out);
00088
00089 /* Encode the specified packet. */
00090 int jpc_enc_encpkt(jpc_enc_t *enc, jas_stream_t *out, unsigned compno, unsigned lvlno,
00091 unsigned prcno, unsigned lyrno);
00092
00093 /* Save the tier-2 coding state. */
00094 void jpc_save_t2state(jpc_enc_t *enc);
00095
00096 /* Restore the tier-2 coding state. */
00097 void jpc_restore_t2state(jpc_enc_t *enc);
00098
00099 /* Initialize the tier-2 coding state. */
00100 void jpc_init_t2state(jpc_enc_t *enc, bool raflag);
00101

```



```

00102 /* Create a packet iterator for the encoder. */
00103 jpc_pi_t *jpc_enc_pi_create(jpc_enc_cp_t *cp, jpc_enc_tile_t *tile);
00104
00105 #endif

```

## 11.62 jpc\_tagtree.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2002 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Tag Tree Library
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_TAGTREE_H

```

```

00071 #define JPC_TAGTREE_H
00072
00073 /*****
00074  * Includes
00075  *****/
00076
00077 #include <stdio.h>
00078
00079 #include "jpc_bs.h"
00080
00081 /*****
00082  * Constants
00083  *****/
00084
00085 /* The maximum allowable depth for a tag tree. */
00086 #define JPC_TAGTREE_MAXDEPTH 32
00087
00088 /*****
00089  * Types
00090  *****/
00091
00092 /*
00093  * Tag tree node.
00094  */
00095
00096 typedef struct jpc_tagtreenode_ {
00097
00098     /* The parent of this node. */
00099     struct jpc_tagtreenode_ *parent_;
00100
00101     /* The value associated with this node. */
00102     int value_;
00103
00104     /* The lower bound on the value associated with this node. */
00105     int low_;
00106
00107     /* A flag indicating if the value is known exactly. */
00108     int known_;
00109
00110 } jpc_tagtreenode_t;
00111
00112 /*
00113  * Tag tree.
00114  */
00115
00116 typedef struct {
00117
00118     /* The number of leaves in the horizontal direction. */
00119     int numleafsh_;
00120
00121     /* The number of leaves in the vertical direction. */
00122     int numleafsv_;
00123
00124     /* The total number of nodes in the tree. */
00125     int numnodes_;
00126
00127     /* The nodes. */
00128     jpc_tagtreenode_t *nodes_;
00129
00130 } jpc_tagtree_t;
00131
00132 /*****
00133  * Functions.
00134  *****/
00135
00136 /* Create a tag tree. */
00137 jpc_tagtree_t *jpc_tagtree_create(int numleafsh, int numleafsv);
00138
00139 /* Destroy a tag tree. */
00140 void jpc_tagtree_destroy(jpc_tagtree_t *tree);
00141
00142 /* Copy data from one tag tree to another. */
00143 void jpc_tagtree_copy(jpc_tagtree_t *dsttree, const jpc_tagtree_t *srctree);
00144
00145 /* Reset the tag tree state. */
00146 void jpc_tagtree_reset(jpc_tagtree_t *tree);
00147
00148 /* Set the value associated with a particular leaf node of a tag tree. */
00149 void jpc_tagtree_setvalue(jpc_tagtree_t *tree, jpc_tagtreenode_t *leaf,
00150     int value);
00151

```

```

00152 /* Get a pointer to a particular leaf node. */
00153 JAS_ATTRIBUTE_PURE
00154 jpc_tagreenode_t *jpc_tagtree_getleaf(jpc_tagtree_t *tree, int n);
00155
00156 /* Invoke the tag tree decoding procedure. */
00157 int jpc_tagtree_decode(jpc_tagtree_t *tree, jpc_tagreenode_t *leaf,
00158     int threshold, jpc_bitstream_t *in);
00159
00160 /* Invoke the tag tree encoding procedure. */
00161 int jpc_tagtree_encode(jpc_tagtree_t *tree, jpc_tagreenode_t *leaf,
00162     int threshold, jpc_bitstream_t *out);
00163
00164 /* Dump a tag tree (for debugging purposes). */
00165 void jpc_tagtree_dump(const jpc_tagtree_t *tree, FILE *out);
00166
00167 #endif

```

## 11.63 jpc\_tsfb.h

```

00001 /*
00002  * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003  *   British Columbia.
00004  * Copyright (c) 2001-2004 Michael David Adams.
00005  * All rights reserved.
00006  */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009  *
00010  * JasPer License Version 2.0
00011  *
00012  * Copyright (c) 2001-2006 Michael David Adams
00013  * Copyright (c) 1999-2000 Image Power, Inc.
00014  * Copyright (c) 1999-2000 The University of British Columbia
00015  *
00016  * All rights reserved.
00017  *
00018  * Permission is hereby granted, free of charge, to any person (the
00019  * "User") obtaining a copy of this software and associated documentation
00020  * files (the "Software"), to deal in the Software without restriction,
00021  * including without limitation the rights to use, copy, modify, merge,
00022  * publish, distribute, and/or sell copies of the Software, and to permit
00023  * persons to whom the Software is furnished to do so, subject to the
00024  * following conditions:
00025  *
00026  * 1. The above copyright notices and this permission notice (which
00027  * includes the disclaimer below) shall be included in all copies or
00028  * substantial portions of the Software.
00029  *
00030  * 2. The name of a copyright holder shall not be used to endorse or
00031  * promote products derived from the Software without specific prior
00032  * written permission.
00033  *
00034  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY

```

```

00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064 /*
00065  * Tree-Structured Filter Bank (TSFB) Library
00066  *
00067  * $Id$
00068  */
00069
00070 #ifndef JPC_TSFB_H
00071 #define JPC_TSFB_H
00072
00073 /*****\
00074  * Includes.
00075  \*****/
00076
00077 #include "jasper/jas_seq.h"
00078 #include "jasper/jas_types.h"
00079
00080 #include "jpc_fix.h"
00081 #include "jpc_qmfb.h"
00082
00083 /*****\
00084  * Constants.
00085  \*****/
00086
00087 #define JPC_TSFB_MAXBANDS      (JPC_TSFB_MAXDEPTH * 3 + 1)
00088 #define JPC_TSFB_MAXDEPTH     32
00089 #define JPC_TSFB_RITIMODE     JPC_QMFB1D_RITIMODE
00090
00091 enum jpc_tsfb_orient {
00092     JPC_TSFB_LL = 0,
00093     JPC_TSFB_LH = 1,
00094     JPC_TSFB_HL = 2,
00095     JPC_TSFB_HH = 3,
00096 };
00097
00098 /*****\
00099  * Types.
00100  \*****/
00101
00102 typedef struct {
00103     int xstart;
00104     int ystart;
00105     int xend;
00106     int yend;
00107     enum jpc_tsfb_orient orient;
00108     int locxstart;
00109     int locystart;
00110     int locxend;
00111     int locyend;
00112     jpc_fix_t synenergywt;
00113 } jpc_tsfb_band_t;
00114
00115 typedef struct {
00116     unsigned numlvls;
00117     const jpc_qmfb2d_t *qmfb;
00118 } jpc_tsfb_t;
00119
00120 /*****\
00121  * Functions.
00122  \*****/
00123
00124 /* Create a TSFB. */
00125 jpc_tsfb_t *jpc_cod_gettsfb(unsigned qmfbid, unsigned numlevels);
00126
00127 /* Destroy a TSFB. */
00128 void jpc_tsfb_destroy(jpc_tsfb_t *tsfb);
00129
00130 /* Perform analysis. */
00131 int jpc_tsfb_analyze(jpc_tsfb_t *tsfb, jas_seq2d_t *x);
00132
00133 /* Perform synthesis. */
00134 int jpc_tsfb_synthesize(jpc_tsfb_t *tsfb, jas_seq2d_t *x);
00135
00136 /* Get band information for a TSFB. */
00137 int jpc_tsfb_getbands(jpc_tsfb_t *tsfb, uint_fast32_t xstart,
00138     uint_fast32_t ystart, uint_fast32_t xend, uint_fast32_t yend,
00139     jpc_tsfb_band_t *bands);

```

```
00140
00141 #endif
```

## 11.64 jpc\_util.h

```
00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef JPC_UTIL_H
00063 #define JPC_UTIL_H
00064
00065 #include "jpc_fix.h"
00066
00067 #include "jasper/jas_seq.h"
00068
00069 /* Parse a comma separated list of real numbers into an array of doubles. */
00070 int jpc_atoaf(const char *s, int *numvalues, double **values);
00071
00072 /* Upsample a sequence. */
```

```
00073 jas_seq_t *jpc_seq_upsample(const jas_seq_t *seq, int n);
00074
00075 /* Convolve two sequences. */
00076 jas_seq_t *jpc_seq_conv(const jas_seq_t *seq0, const jas_seq_t *seq1);
00077
00078 /* Compute the norm of a sequence. */
00079 JAS_ATTRIBUTE_PURE
00080 jpc_fix_t jpc_seq_norm(const jas_seq_t *x);
00081
00082 #endif
```

## 11.65 jpg\_cod.h

```
00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 /*
00063  * JPG Format Library
00064  */
```

```

00065  * $Id$
00066  */
00067
00068 #ifndef JPG_COD_H
00069 #define JPG_COD_H
00070
00071 /*****
00072  * Includes.
00073  *****/
00074
00075 /*****
00076  * Constants.
00077  *****/
00078
00079 #define JPG_MAGIC          0xffd8
00080 #define JPG_MAGICLEN 2
00081
00082 #endif

```

## 11.66 jpg\_enc.h

```

00001 /*
00002  * Copyright (c) 2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY

```

```

00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef JPG_ENC_H
00063 #define JPG_ENC_H
00064
00065 typedef struct {
00066     int numcmpts;
00067     int cmpts[4];
00068 } jpg_enc_t;
00069
00070 #endif

```

## 11.67 jpg\_jpeglib.h

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *
00059  * __END_OF_JASPER_LICENSE__
00060  */

```



```

00061
00062 #ifndef JPG_JPEGLIB_H
00063 #define JPG_JPEGLIB_H
00064
00065 /*****
00066  * Includes.
00067  *****/
00068
00069 #include <stdio.h>
00070 #include "jasper/jas_types.h"
00071
00072 /* Note: The jpeglib.h header file does not include definitions of
00073  FILE, size_t, etc. */
00074 #include <jpeglib.h> /* IWYU pragma: export */
00075
00076 #endif

```

## 11.68 mif\_cod.h

```

00001 /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058  *

```

```

00059  * __END_OF_JASPER_LICENSE__
00060  */
00061
00062 #ifndef MIF_COD_H
00063 #define MIF_COD_H
00064
00065 /*****
00066  * Includes.
00067  *****/
00068
00069 #include "jasper/jas_types.h"
00070
00071 /*****
00072  * Constants.
00073  *****/
00074
00075 #define MIF_MAGIC                0x4d49460a
00076 /* signature */
00077
00078 #define MIF_MAGICLEN            4
00079 /* length of signature in bytes */
00080
00081 /*****
00082  * Types.
00083  *****/
00084
00085 /* Per-component information. */
00086
00087 typedef struct {
00088
00089     int_fast32_t tlx;
00090
00091     int_fast32_t tly;
00092
00093     int_fast32_t width;
00094
00095     int_fast32_t height;
00096
00097     int_fast32_t sampperx;
00098
00099     int_fast32_t samppery;
00100
00101     int_fast16_t prec;
00102
00103     int_fast16_t sgnd;
00104
00105     char *data;
00106
00107 } mif_cmpt_t;
00108
00109 /* MIF header. */
00110
00111 typedef struct {
00112
00113     uint_fast32_t magic;
00114
00115     int numcmpts;
00116
00117     int maxcmpts;
00118
00119     mif_cmpt_t **cmpts;
00120
00121 } mif_hdr_t;
00122
00123 #endif

```

## 11.69 pgx\_cod.h

```

00001  /*
00002  * Copyright (c) 2001-2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  */

```

```

00010 * Copyright (c) 2001-2006 Michael David Adams
00011 * Copyright (c) 1999-2000 Image Power, Inc.
00012 * Copyright (c) 1999-2000 The University of British Columbia
00013 *
00014 * All rights reserved.
00015 *
00016 * Permission is hereby granted, free of charge, to any person (the
00017 * "User") obtaining a copy of this software and associated documentation
00018 * files (the "Software"), to deal in the Software without restriction,
00019 * including without limitation the rights to use, copy, modify, merge,
00020 * publish, distribute, and/or sell copies of the Software, and to permit
00021 * persons to whom the Software is furnished to do so, subject to the
00022 * following conditions:
00023 *
00024 * 1. The above copyright notices and this permission notice (which
00025 * includes the disclaimer below) shall be included in all copies or
00026 * substantial portions of the Software.
00027 *
00028 * 2. The name of a copyright holder shall not be used to endorse or
00029 * promote products derived from the Software without specific prior
00030 * written permission.
00031 *
00032 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 /*
00063  * PGX Format Library
00064  *
00065  * $Id$
00066  */
00067
00068 #ifndef PGX_COD_H
00069 #define PGX_COD_H
00070
00071 /*****
00072  * Includes.
00073  *****/
00074
00075 #include <stdio.h>
00076
00077 #include "jasper/jas_types.h"
00078
00079 /*****
00080  * Constants.
00081  *****/
00082
00083 #define PGX_MAGIC      0x5047
00084 #define PGX_MAGICLEN   2
00085
00086 /*****
00087  * Types.
00088  *****/
00089
00090 typedef struct {

```

```

00091
00092     uint_fast16_t magic;
00093     /* The signature. */
00094
00095     bool bigendian;
00096     /* The byte ordering used. */
00097
00098     bool sgnd;
00099     /* The signedness of the samples. */
00100
00101     uint_fast32_t prec;
00102     /* The precision of the samples. */
00103
00104     uint_fast32_t width;
00105     /* The width of the component. */
00106
00107     uint_fast32_t height;
00108     /* The height of the component. */
00109
00110 } pgx_hdr_t;
00111
00112 /*****
00113  * Functions.
00114  *****/
00115
00116 void pgx_dumphdr(FILE *out, pgx_hdr_t *hdr);
00117
00118 #endif

```

## 11.70 pgx\_enc.h

```

00001 /*
00002  * Copyright (c) 2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL

```

```

00047 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 #ifndef PGX_ENC_H
00063 #define PGX_ENC_H
00064
00065 typedef struct {
00066     int cmpt;
00067 } pgx_enc_t;
00068
00069 #endif

```

## 11.71 pnm\_cod.h

```

00001 /*
00002 * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003 * British Columbia.
00004 * Copyright (c) 2001-2002 Michael David Adams.
00005 * All rights reserved.
00006 */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009 *
00010 * JasPer License Version 2.0
00011 *
00012 * Copyright (c) 2001-2006 Michael David Adams
00013 * Copyright (c) 1999-2000 Image Power, Inc.
00014 * Copyright (c) 1999-2000 The University of British Columbia
00015 *
00016 * All rights reserved.
00017 *
00018 * Permission is hereby granted, free of charge, to any person (the
00019 * "User") obtaining a copy of this software and associated documentation
00020 * files (the "Software"), to deal in the Software without restriction,
00021 * including without limitation the rights to use, copy, modify, merge,
00022 * publish, distribute, and/or sell copies of the Software, and to permit
00023 * persons to whom the Software is furnished to do so, subject to the
00024 * following conditions:
00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE

```

```

00052  * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053  * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064  /*
00065  * Portable Pixmap/Graymap Format Support
00066  *
00067  * $Id$
00068  */
00069
00070  #ifndef PNM_COD_H
00071  #define PNM_COD_H
00072
00073  /*****
00074  * Includes.
00075  *****/
00076
00077  #include "jasper/jas_types.h"
00078
00079  /*****
00080  * Constants.
00081  *****/
00082
00083  /* Magic numbers. */
00084  #define PNM_MAGIC_TXTPBM      0x5031 /* Text Portable BitMap (P1) */
00085  #define PNM_MAGIC_TXTPGM      0x5032 /* Text Portable GrayMap (P2) */
00086  #define PNM_MAGIC_TXTPPM      0x5033 /* Text Portable PixMap (P3) */
00087  #define PNM_MAGIC_BINPBM      0x5034 /* Binary Portable BitMap (P4) */
00088  #define PNM_MAGIC_BINPGM      0x5035 /* Binary Portable GrayMap (P5) */
00089  #define PNM_MAGIC_BINPPM      0x5036 /* Binary Portable PixMap (P6) */
00090  #define PNM_MAGIC_PAM         0x5037 /* PAM (P7) */
00091
00092  /* Type of image data. */
00093  #define PNM_TYPE_INVALID      0
00094  #define PNM_TYPE_PBM          1 /* BitMap */
00095  #define PNM_TYPE_PGM          2 /* GrayMap */
00096  #define PNM_TYPE_PPM          3 /* PixMap */
00097
00098  /* Format of image data. */
00099  #define PNM_FMT_TXT           0 /* Text */
00100  #define PNM_FMT_BIN           1 /* Binary */
00101
00102  #define PNM_MAXLINELEN       79
00103
00104  #define PNM_TUPLETYPE_UNKNOWN 0
00105  #define PNM_TUPLETYPE_MONO    1
00106  #define PNM_TUPLETYPE_GRAY    2
00107  #define PNM_TUPLETYPE_GRAYA   3
00108  #define PNM_TUPLETYPE_RGB     4
00109  #define PNM_TUPLETYPE_RGBA    5
00110
00111  /*****
00112  * Types.
00113  *****/
00114
00115  /* File header. */
00116
00117  typedef struct {
00118
00119      int magic;
00120      /* The magic number. */
00121
00122      int width;
00123      /* The image width. */
00124
00125      int height;
00126      /* The image height. */
00127
00128      int numcmpts;
00129
00130      int maxval;
00131      /* The maximum allowable sample value. */
00132

```

```

00133 #if 0
00134     int tupletype;
00135 #endif
00136
00137     bool sgnd;
00138     /* The sample data is signed. */
00139 } pnm_hdr_t;
00140
00141
00142 /*****
00143  * Functions.
00144  *****/
00145
00146 int pnm_type(uint_fast16_t magic);
00147 /* Determine type (i.e., PGM or PPM) from magic number. */
00148
00149 int pnm_fmt(uint_fast16_t magic);
00150 /* Determine format (i.e., text or binary) from magic number. */
00151
00152 int pnm_maxvaltodepth(uint_fast32_t maxval);
00153 /* Determine depth (i.e., precision) from maximum value. */
00154
00155 #define PNM_ONES(n) \
00156     ((n) < 32) ? ((1UL << (n)) - 1) : 0xffffffffUL
00157 #endif

```

## 11.72 pnm\_enc.h

```

00001 /*
00002  * Copyright (c) 2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00045  * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046  * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047  * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048  * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049  * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE

```

```

00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 #ifndef PNM_ENC_H
00063 #define PNM_ENC_H
00064
00065 typedef struct {
00066     int numcmpts;
00067     int cmpts[4];
00068 } pnm_enc_t;
00069
00070 #endif

```

## 11.73 ras\_cod.h

```

00001 /*
00002 * Copyright (c) 1999-2000 Image Power, Inc. and the University of
00003 *   British Columbia.
00004 * Copyright (c) 2001-2002 Michael David Adams.
00005 * All rights reserved.
00006 */
00007
00008 /* __START_OF_JASPER_LICENSE__
00009 *
00010 * JasPer License Version 2.0
00011 *
00012 * Copyright (c) 2001-2006 Michael David Adams
00013 * Copyright (c) 1999-2000 Image Power, Inc.
00014 * Copyright (c) 1999-2000 The University of British Columbia
00015 *
00016 * All rights reserved.
00017 *
00018 * Permission is hereby granted, free of charge, to any person (the
00019 * "User") obtaining a copy of this software and associated documentation
00020 * files (the "Software"), to deal in the Software without restriction,
00021 * including without limitation the rights to use, copy, modify, merge,
00022 * publish, distribute, and/or sell copies of the Software, and to permit
00023 * persons to whom the Software is furnished to do so, subject to the
00024 * following conditions:
00025 *
00026 * 1. The above copyright notices and this permission notice (which
00027 * includes the disclaimer below) shall be included in all copies or
00028 * substantial portions of the Software.
00029 *
00030 * 2. The name of a copyright holder shall not be used to endorse or
00031 * promote products derived from the Software without specific prior
00032 * written permission.
00033 *
00034 * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00035 * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00036 * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00037 * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00038 * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00039 * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00040 * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00041 * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00042 * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00043 * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00044 * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00045 * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00046 * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.
00047 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00048 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00049 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00050 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00051 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00052 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00053 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,

```



```

00054  * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00055  * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00056  * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00057  * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00058  * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00059  * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00060  *
00061  * __END_OF_JASPER_LICENSE__
00062  */
00063
00064  /*
00065  * Sun Rasterfile Library
00066  *
00067  * $Id$
00068  */
00069
00070  /*****
00071  * Sun Rasterfile
00072  \*****/
00073
00074  #ifndef RAS_COD_H
00075  #define RAS_COD_H
00076
00077  /*****
00078  * Includes.
00079  \*****/
00080
00081  #include "jasper/jas_types.h"
00082
00083  /*****
00084  * Constants.
00085  \*****/
00086
00087  #define RAS_MAGIC      0x59a66a95
00088  #define RAS_MAGICLEN   4
00089
00090  #define RAS_TYPE_OLD   0
00091  #define RAS_TYPE_STD   1
00092  #define RAS_TYPE_RLE   2
00093
00094  #define RAS_MT_NONE    0
00095  #define RAS_MT_EQUALRGB 1
00096  #define RAS_MT_RAW     2
00097
00098  /*****
00099  * Types.
00100  \*****/
00101
00102  /* Sun Rasterfile header. */
00103  typedef struct {
00104
00105      int_fast32_t magic;
00106      /* signature */
00107
00108      int_fast32_t width;
00109      /* width of image (in pixels) */
00110
00111      int_fast32_t height;
00112      /* height of image (in pixels) */
00113
00114      int_fast32_t depth;
00115      /* number of bits per pixel */
00116
00117      int_fast32_t length;
00118      /* length of image data (in bytes) */
00119
00120      int_fast32_t type;
00121      /* format of image data */
00122
00123      int_fast32_t maptype;
00124      /* colormap type */
00125
00126      int_fast32_t maplength;
00127      /* length of colormap data (in bytes) */
00128
00129  } ras_hdr_t;
00130
00131  #define RAS_CMAP_MAXSIZE 256
00132
00133  /* Color map. */
00134  typedef struct {

```

```

00135
00136     int len;
00137     /* The number of entries in the color map. */
00138
00139     int data[RAS_CMAP_MAXSIZE];
00140     /* The color map data. */
00141
00142 } ras_cmap_t;
00143
00144 /*****
00145  * Macros.
00146  *****/
00147
00148 #define RAS_GETBLUE(x)  (((x) >> 16) & 0xff)
00149 #define RAS_GETGREEN(x) (((x) >> 8) & 0xff)
00150 #define RAS_GETRED(x)  ((x) & 0xff)
00151
00152 #define RAS_BLUE(x)     (((x) & 0xff) << 16)
00153 #define RAS_GREEN(x)   (((x) & 0xff) << 8)
00154 #define RAS_RED(x)     ((x) & 0xff)
00155
00156 #define RAS_ROWSIZE(hdr) \
00157     (((hdr)->width * (hdr)->depth + 15) / 16) * 2)
00158 #define RAS_ISRGB(hdr) \
00159     ((hdr)->depth == 24 || (hdr)->depth == 32)
00160
00161 #define RAS_ONES(n) \
00162     (((n) == 32) ? 0xffffffffUL : ((1UL << (n)) - 1))
00163
00164 #endif

```

## 11.74 ras\_enc.h

```

00001 /*
00002  * Copyright (c) 2002 Michael David Adams.
00003  * All rights reserved.
00004  */
00005
00006 /* __START_OF_JASPER_LICENSE__
00007  *
00008  * JasPer License Version 2.0
00009  *
00010  * Copyright (c) 2001-2006 Michael David Adams
00011  * Copyright (c) 1999-2000 Image Power, Inc.
00012  * Copyright (c) 1999-2000 The University of British Columbia
00013  *
00014  * All rights reserved.
00015  *
00016  * Permission is hereby granted, free of charge, to any person (the
00017  * "User") obtaining a copy of this software and associated documentation
00018  * files (the "Software"), to deal in the Software without restriction,
00019  * including without limitation the rights to use, copy, modify, merge,
00020  * publish, distribute, and/or sell copies of the Software, and to permit
00021  * persons to whom the Software is furnished to do so, subject to the
00022  * following conditions:
00023  *
00024  * 1. The above copyright notices and this permission notice (which
00025  * includes the disclaimer below) shall be included in all copies or
00026  * substantial portions of the Software.
00027  *
00028  * 2. The name of a copyright holder shall not be used to endorse or
00029  * promote products derived from the Software without specific prior
00030  * written permission.
00031  *
00032  * THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS
00033  * LICENSE. NO USE OF THE SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER
00034  * THIS DISCLAIMER. THE SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS
00035  * "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
00036  * BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
00037  * PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO
00038  * EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL
00039  * INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING
00040  * FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT,
00041  * NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION
00042  * WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. NO ASSURANCES ARE
00043  * PROVIDED BY THE COPYRIGHT HOLDERS THAT THE SOFTWARE DOES NOT INFRINGE
00044  * THE PATENT OR OTHER INTELLECTUAL PROPERTY RIGHTS OF ANY OTHER ENTITY.

```

```
00045 * EACH COPYRIGHT HOLDER DISCLAIMS ANY LIABILITY TO THE USER FOR CLAIMS
00046 * BROUGHT BY ANY OTHER ENTITY BASED ON INFRINGEMENT OF INTELLECTUAL
00047 * PROPERTY RIGHTS OR OTHERWISE. AS A CONDITION TO EXERCISING THE RIGHTS
00048 * GRANTED HEREUNDER, EACH USER HEREBY ASSUMES SOLE RESPONSIBILITY TO SECURE
00049 * ANY OTHER INTELLECTUAL PROPERTY RIGHTS NEEDED, IF ANY. THE SOFTWARE
00050 * IS NOT FAULT-TOLERANT AND IS NOT INTENDED FOR USE IN MISSION-CRITICAL
00051 * SYSTEMS, SUCH AS THOSE USED IN THE OPERATION OF NUCLEAR FACILITIES,
00052 * AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL
00053 * SYSTEMS, DIRECT LIFE SUPPORT MACHINES, OR WEAPONS SYSTEMS, IN WHICH
00054 * THE FAILURE OF THE SOFTWARE OR SYSTEM COULD LEAD DIRECTLY TO DEATH,
00055 * PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE ("HIGH
00056 * RISK ACTIVITIES"). THE COPYRIGHT HOLDERS SPECIFICALLY DISCLAIM ANY
00057 * EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HIGH RISK ACTIVITIES.
00058 *
00059 * __END_OF_JASPER_LICENSE__
00060 */
00061
00062 #ifndef RAS_ENC_H
00063 #define RAS_ENC_H
00064
00065 typedef struct {
00066     int numcmpts;
00067     int cmpts[4];
00068 } ras_enc_t;
00069
00070 #endif
```

